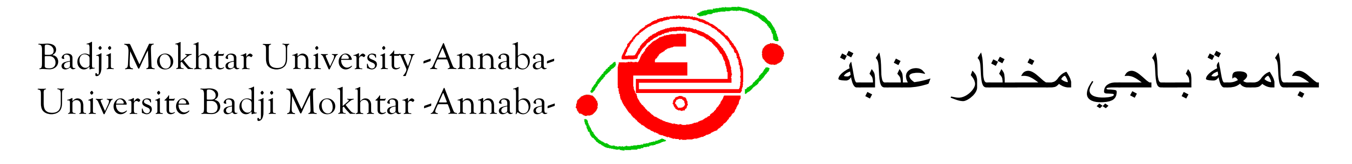
وزارة التعليم العالي و البحث العلمي



Faculté : Sciences de l'Ingénieur **Année 2010**

Département : Informatique

MEMOIRE

Présenté en vue de l'obtention du Diplôme de **Magister**

Une implémentation d'un environnement à base de Grid

Option

Texte Parole et Image

Par

ABDOUNE Leila

DIRECTEUR DE MEMOIRE: Mokhtar SELLAMI Professeur Université de Annaba

DEVANT LE JURY

PRESIDENTE :	Hassina SERIDI	MC	Université de Annaba
Examinateurs :	Nadir FARAH	MC	Université de Annaba
	Habiba BELLEILI	MC	Université de Annaba

ملخّص

القريد هو نموذج جديد للحصول على الموارد الموزعة وغير المتجانسة؛ ومع ظهور معيار OGSA، أصبح البنية التحتية للعديد من التطبيقات والمشاريع الكبيرة. التعليم الإلكتروني هو واحد من أهم المجالات حيث أثبت القريد مساهمته. هذا البحث يهدف إلى تقديم بنية لبيئة تعاونية في مجال التعليم الإلكتروني على أساس تكنولوجيات القريد من خلال محاولة استغلال موارد الجامعات والمؤسسات الغير مستغلة بالقدر الكافي من جهة ، و دعم نظم التعلم الموجودة من جهة أخرى. لذلك، بيئتنا التعاونية ستكون مكانا لاجتماع المجموعات المعزولة من أجل التعلم باستخدام منصات مختلفة، وإعطاء الفرصة لهم للعمل جماعيا في مجال التعليم والتعلم، وهذا عن طريق تقاسم الموارد الموزعة وغير المتجانسة، وتقاسم محتوى التعلم والتعاون بين أعضاء من مختلف المجموعات. يبدأ هذا البحث بشرح أسباب اعتماد القريد ، ثم يسلط الضوء على أهمية القريد في التعلم الإلكتروني والتعاون وباختصار ، لدى القريد مميزات و وظائف رئيسية التي بإمكانها أن تلبي متطلبات بيئات تعاونية ؛ ولكن مع هذا يبقى في حاجة إلى استخدام تقنيات أخرى مهمة ، مثل الشبكة الدلالية.

كلمات البحث : القريد ، التعليم الإلكتروني ، التعاون ، أوقسا، غلوبوس

ABSTRACT

The Grid system is a new paradigm of accessibility to the distributed and heterogeneous resources, and with the appearance of OGSA standard, it becomes an infrastructure for numerous applications; and the big world projects now adopt it. The E-learning is one of the most important fields in which the Grid has shown its contribution. In this regard, the research paper aims at presenting an architecture of a collaboration environment in the area of E-learning, basing on the Grid technologies. On the one hand, our goal is to exploit the underused resources in universities and institutions, and on the other hand, to back up the existing learning systems. Therefore, our collaboration environment would be a meeting space of the isolated learning communities that use different platforms. Here, we enable them to work together in the learning/teaching fields via the sharing of distributed and heterogeneous resources, the sharing of the learning contents, and the collaboration between the different members of communities. The paper explains the reasons that lie behind the adoption of the Grid system, and it also highlights the importance of this system for both the E-learning and collaboration. In sum, the Grid has some characteristics and chief functionalities that can meet the needs of the collaboration environments; yet, we can assert that despite its richness, the use of other technologies proved to be necessary, such as the semantic Grid

Keywords: Grid Computing, E-learning, collaboration, OGSA, Globus Toolkit.

RESUME

Le Grid est un nouveau paradigme d'accessibilité aux ressources distribuées et hétérogènes, et avec l'apparition du standard OGSA, il est devenu une infrastructure pour de nombreuses applications ; et les grands projets du monde l'utilisent. Le E-learning est l'un des domaines les plus importants où le Grid a montré sa contribution. Ce mémoire a pour but de présenter une architecture d'un environnement de collaboration dans le domaine du E-learning en s'appuyant sur les technologies de Grid en essayant d'exploiter les ressources sous utilisées des universités et instituions, d'une part, et de soutenir les systèmes d'apprentissage existants, d'autre part. Par conséquent, notre environnement de collaboration sera un lieu de rencontre des communautés d'apprentissage isolées qui utilisent des plateformes différentes en leur donnant la possibilité de travailler ensemble dans le domaine d'enseignement et d'apprentissage, ceci via le partage de ressources distribuées et hétérogènes, le partage de contenus d'apprentissage, et la collaboration entre les membres des différentes communautés. Cette recherche explique en premier lieu, les raisons de l'adoption du Grid, ensuite met en exergue l'apport de Grid pour le E-learning et la collaboration. En somme, le Grid dispose des caractéristiques et des fonctionnalités principales qui peuvent répondre aux exigences des environnements de collaboration, il est toutefois nécessaire de souligner que malgré la richesse de ce système, l'appel à d'autres technologies –tel que le Grid sémantique– s'avère important.

Mots clés: Grille de calcul, E-learning, collaboration, OGSA, Globus Toolkit.

DEDICACES

Ce mémoire est dédié à mes parents, qui m'ont toujours poussé et motivé dans mes études. Sans eux, je n'aurais certainement pas fait d'études longues. Ce mémoire représente donc l'aboutissement du soutien et des encouragements qu'ils m'ont prodigués tout au long de ma scolarité. Qu'ils en soient remerciés par cette trop modeste dédicace.

A mes frères et soeurs,
A mon fiancé,
A tous mes amis,
A tous ceux que j'aime,
Je dédie ce travail.

REMERCIEMENTS

Il m’est très agréable de réserver cette page comme un témoin de reconnaissance à toutes les personnes qui m’ont soutenu et aidé pour réaliser ce travail. Je remercie d’abord Allah le tout puissant, qui m’a donné la foi, la force et la patience pour aller jusqu’au bout de ce travail.

Mes vifs et sincères remerciements à mon encadreur Professeur Mokhtar SELLAMI : Professeur en informatique à l’université BADJI MOKHTAR ANNABA, qui m’a fait le grand honneur d’accepter de me diriger dans ce modeste travail, pour ses précieux conseils, sa collaboration et sa modestie exemplaire. Votre rigueur scientifique et vos qualités pédagogiques m’ont aidé tout au long de la réalisation de ce travail, je vous prie de voir dans ce mémoire l’expression de ma respectueuse gratitude.

Je voudrais remercier madame SERIDI Hassina, maître de conférence à l’université de Annaba de m’avoir fait l’honneur de présider mon jury.

Je veux également exprimer toute ma gratitude à madame BELLEILI et à monsieur FARAH maîtres de conférence à l’université de Annaba qui ont eu la grande gentillesse de mettre leur savoir au service de mon travail en acceptant de le juger.

Je tiens à exprimer également mes chaleureux remerciements aux responsables du Programme SARIMA qui m’ont financé et facilité la réalisation de mon stage en Tunisie.

Je remercie aussi le professeur Yahia SLIMANI et Mr Moez BEN HADJ HMIDA de m’avoir accueilli en stage au sein du département des sciences de l’informatique, université de TUNIS EL MANAR, et qui m’ont énormément aidé.

Mes plus vifs remerciements à Monsieur Hadi BOUTARFA professeur d’anglais à l’université d’Annaba pour son aide précieuse lors de la lecture de ce manuscrit.

Cette page n’aurait probablement pas pu s’écrire sans l’appui moral des membres de ma famille. Je les remercie tous, particulièrement mon père, ma mère, mes frères, mes sœurs et mon fiancé pour l’amour qu’ils m’ont apporté tous les jours, sans oublier mes amis pour leur aide et soutien moral aux moments où tout allait mal, à vous tous merci.

Enfin, Que toute personne ayant contribué de près ou de loin à la réalisation de ce travail par une quelconque forme de contribution trouve ici le témoignage de ma plus profonde reconnaissance.

LISTE DES TABLEAUX

TABLE DES FIGURES

TABLE DES MATIERES

INTRODUCTION GENERALE

1. INTRODUCTION

Le E-learning est un sujet d'intérêt croissant au cours des dernières années, par conséquent différentes plateformes et systèmes de E-learning ont été développés **[PV03]**, qui différent par la nature de leurs architectures ; architectures client/serveur, architectures pair à pair et architectures orientées services. Avec l'évolution des technologies d'information et de communication le E-learning est devenu une méthode d'apprentissage très employée dans différents secteurs ; Les grandes entreprises en association avec la gestion des connaissances pour former et actualiser les connaissances de ses employés de manière plus rapide qu'à l'ordinaire, les universités qui offrent des alternatives de formation à distance en formant des universités virtuelles qui donnent accès à l'éducation autrement pas possible pour certains étudiants en raison des contraintes de temps ou d'espace ou parce qu'il est trop coûteux et finalement, tout particulier qui voudrait entamer un apprentissage en ligne.

La nature distribuée des environnements d'apprentissage, leur héterogenieté (des apprenants ou utilisateurs qui se situent dans des endroits géographique différents, des ressources distribuées et hétérogènes) et le volume important des bases de données nécessitent des systèmes très puissants pour surmonter les difficultés de ces environnements.

La collaboration joue un rôle important dans le processus d'apprentissage et plus précisément dans les environnements de E-learning ; elle permet l'échange des connaissances, du savoir faire, d'idées sur un sujet particulier, et donne aux différents acteurs (apprenants, enseignants, administrateurs,…) la possibilité de travailler en commun pour suivre des activités d'apprentissage, création de cours, évaluation des résultats, etc. Les utilisations des technologies de l'information et de la communication sont de plus en plus nombreuses et offrent à leurs utilisateurs de plus grandes possibilités de collaboration et d'amélioration de leurs performances. Aujourd'hui, avec l'émergence des technologies de Grid s'ouvrent de nouvelles autres perspectives qui pallient aux limites des services web actuels; les utilisateurs à partir de leur poste de travail peuvent exécuter des applications sur un ensemble de ressources distribuées et hétérogènes. Avec la maturité de la technologie de Grid et l'apparition de l'architecture de

services Grid OGSA, son domaine d'application est devenu de plus en plus vaste notamment l'éducation et la collaboration.

2. PROBLEMATIQUE

Aujourd'hui, de nombreuses recherches ont pour objectif de fournir un environnement de collaboration dans plusieurs domaines comme le commerce, la santé et l'éducation où la collaboration humaine est exigée. Notre objectif dans ce mémoire est la mise à disposition d'un environnement de collaboration dans le domaine du E-learning. Cependant, plusieurs travaux et projets ont vu le jour dans ce domaine là, plusieurs environnements de collaboration ont été proposés, implémentés et validés et permettent de faire réussir l'apprentissage collaboratif entre un groupe d'apprenants en leur fournissant les outils nécessaires pour acquérir des connaissances malgré les contraintes de temps et de l'espace. Pourquoi alors proposer un environnement de collaboration ? A qui s'adresse t'il ?

Récemment, avec l'évolution de la technologie d'information et de communication (TIC) et la disponibilité des outils de travail collaboratif, une nouvelle forme de travail collectif est apparue, c'est les communautés virtuelles d'apprentissage appelées VLCs **[DPC03]** ; ces communautés représentent des communautés réelles, elles comprennent de vraies personnes et des enjeux importants, elle est dite virtuelle car les interactions entre les différents membres ne se font pas face à face mais à l'aide des TIC et par le biais d'Internet. Les VLCs nécessitent un environnement de collaboration qui va leur permettre de travailler ensemble, de partager les ressources et même les enseignants.

Mettre à disposition des communautés d'apprentissage un environnement de collaboration ne veut pas dire assurer les activités d'apprentissage via l'environnement mais le rôle de ce dernier est d'assurer l'agrégation des ressources distribuées et leur partage, et rendre possible la communication et la collaboration. Nous supposons alors que chaque communauté a sa propre plateforme d'apprentissage (LMS), cependant les fonctions et les objets d'apprentissage qu'elles fournissent dépendent de la plateforme et par conséquent ne peuvent être échangés ou réutilisés dans d'autres plateformes. Le problème posé d'héterogenieté des contenus ainsi que des fonctions va rendre la collaboration entre les différents utilisateurs de chaque plateforme une tache très complexe.

Nous constatons ici la complexité du problème posé et pour le simplifier nous allons mettre l'accent sur les questions suivantes :

- Comment rendre possible la communication des communautés possédant des LMSs différents ?

- Comment permettre aux communautés de partager des ressources distribuées et hétérogènes ?
- Comment rendre possible la collaboration entre les membres des communautés différentes ?
- Comment gérer les communautés ?

Pour pouvoir leur proposer un environnement de collaboration le plus adéquat en terme de leur nature et besoins, nous nous sommes basés sur les deux questions qui suivent :

- Quelle est la technologie qui répond le plus à leurs exigences ?
- Quels sont les mécanismes qu'on va utiliser pour renforcer la collaboration entre les entités humaines qui composent les différentes communautés ?

3. CONTRIBUTION

L'objectif principal de Grid est le partage coordonné de ressources et la résolution de problèmes au sein d'organisations virtuelles dynamiques et multi-institutionnelles **[FKT01]**. En effet, le Grid traite trois catégories de problèmes : Les problèmes axés sur le calcul, où le Grid leur forme une solution en combinant de puissantes ressources de calcul pour traiter des problèmes qui ne pourraient pas être résolus sur un système unique ou pour le faire beaucoup plus rapidement. Les problèmes axés sur les données, dans ce type d'application le Grid sert à collecter, mettre en mémoire et analyser des données qui seront conservées dans des dépôts, des bibliothèques numériques et des bases de données géographiquement réparties. Finalement, les problèmes axés sur le travail en commun, appelés aussi applications coopératives, ont principalement pour objet d'améliorer les interactions entre des personnes en essayant de rapprocher des individus ou des collectivités aux fins de collaboration.

Le troisième type d'application est celui que nous venons de présenter dans la problématique, où les personnes qui forment notre environnement de collaboration sont géographiquement distribuées, et ont un objectif commun qui est le transfert et l'atteinte de connaissances. Pour cette raison, nous avons utilisé le Grid afin de résoudre les problèmes posés par les environnements de collaboration et plus précisément les problèmes techniques, tels que le partage de ressources et l'assurance des capacités de traitement et de stockage nécessaires. Par conséquent, la première motivation pour le choix de Grid est sa capacité de partage de ressources distribuées et hétérogènes qui présente un des premiers points à assurer dans un environnement de collaboration.

La deuxième motivation est le fait que Grid est considéré comme la technologie leader à l'appui et la gestion des organisations virtuelles, ce qui reflète l'objectif de notre environnement.

Dans ce mémoire, nous présentons une architecture d'un environnement de collaboration pour le E-learning en se basant sur la technologie de Grid, en essayant de bénéficier des potentialités que le Grid offre soit pour le E-learning ou la collaboration.

4. ORGANISATION DU DOCUMENT

Le mémoire s'articule autour de deux parties principales, plus une introduction, une conclusion générale et une annexe. La première partie, présente un état de l'art qui est organisé en trois principaux chapitres. La deuxième partie, présente notre proposition qui fait l'objet d'un quatrième chapitre. En réalité, l'ordre des chapitres de la partie "état de l'art" choisi dans ce mémoire n'est pas fait au hasard, mais c'est du à la nature des informations présentées dans chacun de ces chapitres et les relations existantes entre eux; il est important de présenter d'abord la technologie de Grid puis passer à l'intergiciel qui l'implémente. Vu que le chapitre E-learning contient une partie synthèse des travaux qui expose un ensemble de travaux qui se basent sur le Grid, il est difficile de comprendre leur principe sans avoir déjà pris une idée sur c'est quoi le Grid. Enfin, l'annexe présente une partie importante de ce mémoire qui décrit en détail les étapes nécessaires pour la mise en œuvre d'un nœud de Grid.

Le contenu des différents chapitres est résumé ci-dessous :

Chapitre 1 - Les grilles de calcul Présente la technologie de Grid, son principe, domaines d'application, standards utilisés, et les intergiciels existants.

Chapitre 2 - l'intergiciel Globus Nous aide à bien comprendre le principe de Grid et les services qu'il offre qui sont décrits dans le premier chapitre. Ce chapitre contient une description détaillée des services offerts par l'intergiciel Globus et leur principe de fonctionnement.

Chapitre 3 - E-learning et collaboration Présente le E-learning et ses concepts de base, nous introduisons également l'apprentissage collaboratif et les communautés virtuelles d'apprentissage. Finalement, nous terminerons ce chapitre par une synthèse de quelques travaux.

Chapitre 4 - Architecture proposée Afin de répondre à notre problématique qui s'agit d'exploiter les puissances de Grid pour donner une solution au problème de collaboration qui est un problème posé dans de nombreux domaines et plus précisément le domaine de E-learning, nous consacrons ce chapitre pour la proposition d'une architecture qui facilite la collaboration entre différentes communautés qui possèdent chacune son propre LMS.

PARTIE 1

Etat de l'art

Dans l'objectif de mieux cerner le contexte de notre travail, cette partie présente tous les concepts de base liés aux technologies de Grid, les environnements de E-learning ainsi que la collaboration au sein et entre les communautés virtuelles d'apprentissage, qui sont organisés en trois principaux chapitres :

- ***Chapitre 1*** : *Les grilles de calcul*
- ***Chapitre 2*** *: L'intergiciel Globus*
- ***Chapitre 3*** *: E-learning et collaboration*

CHAPITRE 1 :

LES GRILLES DE CALCUL

***Le concept du Grid computing devrait aboutir un jour à la virtualisation de l'ensemble de l'infrastructure informatique mondiale* [Glo03].**

Le Grid ; ce nouveau paradigme d'accessibilité aux ressources distribuées, inspiré de la grille d'électricité, est devenu aujourd'hui une infrastructure de base pour la plupart des applications, facilitant le partage coordonné des ressources et offrant des capacités de traitement et de stockage nécessaires pour l'exécution de ces dernières.

Dans ce chapitre nous exposons le concept de Grid, ses standards et domaines d'applications. Par la suite, nous présentons l'architecture des services Grid OGSA, qui présente une série de spécifications techniques par lesquelles on définit une infrastructure pour intégrer et gérer les services à l'intérieur d'une organisation virtuelle, distribuée et dynamique. Enfin, nous présentons un panorama rapide sur quelques intergiciels qui sont en effet les implémentations de l'architecture OGSA.

1.1. INTRODUCTION

L'informatique est en évolution continue, chaque jour de nouvelles capacités se multiplient, des ordinateurs individuels avec de grandes capacités de stockage et de calcul, des réseaux de haut débit, des grappes de calcul très puissantes, etc., mais de l'autre coté, les besoins des applications de différents domaines augmentent plus rapidement et dépassent les capacités de ces ressources lorsqu'elles sont utilisées de façon individuelle.

L'objectif de la grille de calcul est d'exploiter l'ensemble des ressources informatiques (supercalculateurs, serveurs de bases de données ou stations de travail, etc.) hétérogènes et distribuées - appartenant à des personnes et des institutions différentes et reliées par un réseau la plupart du temps Internet- pour construire un système de calcul virtuel en permettant le partage de ces ressources et en virtualisant la manière d'y accéder.

L'infrastructure de Grid s'est évoluée par l'adoption de nouvelles technologies, tels que les Web services et par conséquent son domaine d'application est devenu plus large. Aujourd'hui, les grands projets du monde l'utilisent et bénéficient des capacités qu'elle offre en terme de puissance de calcul et des capacités de stockage et la possibilité de faire réussir le travail collaboratif entre des entités (individus et ressources hétérogènes) qui sont géographiquement distribuées.

Ce chapitre présente les concepts liés à la grille de calcul, les motivations de cette technologie ainsi que les applications visées, et enfin une conclusion sur les points abordés tout au long de ce chapitre.

1.2. PRESENTATION DE GRID

Dans cette section nous allons décrire l'origine de Grid, ses objectifs et l'évolution de sa définition avec l'évolution de la catégorie des problèmes à résoudre.

1.2.1. L'ORIGINE DE LA TECHNOLOGIE DE GRID

Le concept de "grille de calcul"ou (Grid[1] en anglais) a été introduit la première fois en 1999 par Ian Foster et Carl Kesselman dans leur livre "*The Grid: Blueprint for a new Computing Infrastructure*" **[FK99a]**, pour décrire une infrastructure de calcul distribuée pour la science avancée et l'ingénierie. Ensuite, cette infrastructure a évoluée dans son extension et application pour résoudre d'autres catégories de problèmes.

[1]Tout au long de ce mémoire nous utilisons le terme Grid pour désigner la grille de calcul

Le concept de Grid vient de l'analogie avec le réseau de distribution d'électricité appelé "*electrical power grid*" en anglais. En effet, le Grid fonctionne comme un réseau électrique où le consommateur se fournit sans se préoccuper de savoir où et par qui est produite l'énergie.

Les technologies et les concepts de Grid ont été initialement développés pour permettre le partage de ressources dans les collaborations scientifiques. Un thème commun sous-jacent de ces différentes modalités d'usage est le besoin *d'un partage coordonné de ressources et la résolution de problèmes au sein d'organisations virtuelles dynamiques et multi-institutionnelle*s **[FKT01]**.

1.2.2. DEFINITION

Plusieurs définitions ont été proposées, nous présenterons ici la première définition donnée par Ian Foster **[FK99a]** :

La grille de calcul est une infrastructure matérielle et logicielle fournissant un accès fiable, cohérant, à taux de pénétration élevé et bon marché à des capacités de traitement et de calcul.

A partir de cette définition nous pouvons tirer les points suivants :

- L'objectif principal de Grid est d'assurer et de mutualiser des ressources qui peuvent être des cycles processeurs, des données, des espaces de stockage. Afin de réaliser cet objectif nous aurons besoin d'une infrastructure matérielle qui sert à interconnecter ces ressources et d'une infrastructure logicielle permettant le contrôle et la gestion de ce matériel.
- Le besoin d'un accès fiable aux ressources est fondamental ce qui implique l'assurance aux utilisateurs d'un service prévisible, constant et de haute performance lors de l'accès aux ressources de Grid. Les performances peuvent désigner la bande passante réseau, le temps de latence, la puissance de calcul, la sécurité, etc.
- Le besoin d'un service cohérent est le deuxième aspect fondamental. La cohérence suggère la présence de services standards, accessibles via des interfaces standards et opérant selon des paramètres standards, tout comme dans la grille d'électricité. Un des grands défis de la conception de tels standards est l'encapsulation de l'hétérogénéité sans compromettre l'exécution de haute performance.
- Un taux de pénétration élevé permet de garantir la facilité d'accès aux services par une large population.
- Finalement l'infrastructure doit offrir un accès relativement bon marché au regard des bénéfices qu'elle peut apporter. L'aspect bon marché est très important d'un point de vue viabilité économique.

Par la suite, en 2000 dans **[FKT01]** cette définition a été raffinée pour adresser les problèmes politiques et sociales en déclarant que le Grid est concerné par le "*partage coordonné de ressources et la résolution de problèmes au sein d'organisations virtuelles dynamiques et multi-institutionnelles*".

- Le concept clé est l'aptitude à négocier les arrangements de partage de ressources entre les participants (producteurs et consommateurs) et ensuite utiliser le pool des ressources résultant pour certains buts.
- Le partage désigné ici n'est pas l'échange de fichiers, mais plutôt l'accès direct aux ressources (ordinateurs, logiciels, données, etc.).
- Ce partage est nécessairement hautement contrôlé avec les producteurs et les consommateurs de ressources qui définit clairement et attentivement ce qui est partagé, qui est autorisé à partager, et les conditions sous lesquelles le partage se produit.
- Un ensemble d'individus et/ou d'institutions définis par les règles d'un tel partage forment ce que nous appelons une organisation virtuelle.

Vu l'évolution technologique dans ces dernières années, le Grid a évolué par conséquent, et son domaine d'application s'est élargi. Le Grid est exploité pour résoudre un autre type de problème c'est les problèmes commerciaux, ce qui a fait évoluer ce concept et par conséquent, des changements ont été apportées sur sa définition. Foster a donné une définition finale dans **[Fos02]**, où il définit le Grid à l'aide des trois points suivants :

1. *Des ressources informatiques coordonnées dont leur administration n'est pas centralisée.* le Grid intègre et cordonne des ressources et des utilisateurs qui appartiennent à des domaines administratifs différents ;
2. *Le Grid utilise des interfaces et des protocoles standards, ouverts et universels ;*
3. *Dans le but de délivrer des qualités de service non triviales.* le Grid a pour objectif de délivrer des qualités de service reliées par exemple au temps de réponse, au débit, la disponibilité et la sécurité, et/ou la co-allocation de multiples types de ressources pour satisfaire les besoins complexes des utilisateurs de telle sorte que l'utilité du système entier est plus grande que la somme de ses parties.

1.2.3. LES ORGANISATIONS VIRTUELLES (OV)

L'organisation virtuelle (OV[2]) constitue le point clé de Grid, c'est un ensemble dynamique d'individus, de groupes et d'institutions ou d'organisations qui partagent des ressources sous une

[2] Nous utilisons le terme OV pour désigner "Organsation Vvirtuelle" dans tout le document.

manière contrôlée en collaborant afin d'atteindre un objectif commun **[FKT01]**. Le partage de ressources dans une OV est hautement contrôlé entre les producteurs et consommateurs des ressources en définissant les ressources à partager, qui a le droit de les partager et sous quelles conditions.

1.3. DOMAINES D'APPLICATION

Il existe cinq grandes classes majeures des applications de Grid telles qu'elles sont définies par Foster et Kesselman dans **[FK99b]** :

1.3.1. CALCUL DISTRIBUE (DISTRIBUTED SUPERCOMPUTING)

Agréger les ressources de calcul importantes afin de faire face à des problèmes qui ne peuvent pas être résolus sur un seul système. Le calcul distribué est un excellent exemple sur les applications qui peuvent bénéficier de Grid, les ressources peuvent être des supercalculateurs ou des stations de travail d'une université. Nous citons comme exemple la simulation distribuée dans la météorologie et la simulation des processus physiques complexes. Cette classe est caractérisée par un nombre important de problèmes qui nécessitent de grandes puissances CPU, stockage, etc.

1.3.2. CALCUL HAUT-DÉBIT (HIGH-THROUGH OUT COMPUTING)

Dans le calcul à haut débit le Grid est utilisé pour l'ordonnancement d'un grand nombre de tâches indépendantes ou faiblement couplées dans le but d'utiliser les cycles processeurs inutilisés, ces tâches peuvent avoir un objectif commun (résoudre le même problème) comme ils peuvent avoir différents objectifs (des problèmes très différents). Il existe plusieurs exemples nous citons quelques uns :

- Le système Condor **[Con09]** de l'université de Wisconsin-Madison peut être utilisé pour gérer des clusters et doté de mécanismes pour la récupérations des cycles processeurs inutilisés des stations de travail, il permet d'ordonnancer une série de tâches parallèles soumise par l'utilisateur en choisissant où et quand ces tâches vont être exécutées, gérer et suivre leurs évolution et informer les utilisateurs de la fin de l'exécution. Ce système a été utilisé par une centaine d'organisations académiques, industrielles et gouvernementales, ce nombre important de ressources est utilisé pour des études aussi diverses et variées que la simulation moléculaire et la résolution des problèmes mathématiques comme NUG30, etc.
- Le constructeur de microprocesseurs AMD a utilisé les techniques de calcul à haut débit lors de la phase de conception de ses modèles K6 et K7.

Cette classe est caractérisée par l'exploitation des capacités de nombreuses ressources qui sont lentes lorsque elles sont utilisées individuellement pour augmenter le débit global.

1.3.3. CALCUL A LA DEMANDE (ON-DEMAND COMPUTING)

Ce genre d'applications utilise le Grid afin de satisfaire les besoins en ressources (temps de calcul, logiciels, données, équipement spécifique, etc.) qui ne peuvent pas être assurés en local, l'utilisation de ces ressources se caractérise par le fait qu'elle est temporaire et dont la possession permanente n'est pas rentable. Contrairement aux applications de calcul distribué, le point le plus important est le rapport performance-prix plutôt que les performances absolues. L'objectif est de réduire la complexité des fonctionnements distribués et de baisser les coûts **[Lef03]**. Cette classe est caractérisée par l'intégration des ressources distantes pour achever des traitements locaux, pour une période de temps limitée.

1.3.4. TRAITEMENT MASSIF DE DONNEES (DATA-INTENSIVE COMPUTING)

Les applications de traitement intensif de données sont basées sur l'extraction de nouvelles informations à partir des données qui sont maintenues dans des bases de données et des bibliothèques numériques distribuées géographiquement. Le processus d'extraction lui même est une tâche qui nécessite la bande passante et la puissance de calcul.

Le traitement massif de données est concerné aussi par la capture, l'analyse, le management et la compréhension des données à des volumes et taux qui dépassent les capacités des technologies actuelles **[Dic09]**.

Comme exemple d'application, nous pouvons citer la physique des hautes énergies qui produit d'énormes quantités de données par jour, la prévision météorologique à long terme et les programmes d'observation terrestre, etc. Les applications de traitement intensif de données apparaissent simultanément dans les domaines scientifiques (L'astronomie, la physique des hautes énergies et la biologie comme la génomique) et commerciaux (services financiers, le commerce). **[FK04]**. Cette classe est caractérisée par la synthèse de nouvelles informations pour un nombre important de sources de données.

1.3.5. CALCUL COLLABORATIF (COLLABORATIVE COMPUTING)

L'objectif des applications collaboratives et de permettre et favoriser les interactions entre humains. Ces applications sont souvent structurées en terme des espaces virtuels partagées. Les applications collaboratives ont presque les mêmes caractéristiques que les autres types cités ci-dessus, du fait que la plupart d'entre elles permettent l'utilisation partagée de ressources de calcul.

Aujourd'hui, le calcul collaboratif est utilisé pour un grand nombre d'applications et domaines à savoir la découverte de connaissances, les environnements collaboratifs et le E-learning. Cette classe est caractérisée par le support de la communication ou du travail collaboratif entre plusieurs participants.

Aujourd'hui, avec la maturité et l'évolution de la technologie des Grids, les grands projets du monde l'utilisent et bénéficient des capacités qu'elle offre en terme de puissance de calcul et des capacités de stockage et la possibilité de faire réussir le travail collaboratif entre des entités (individus et ressources hétérogènes) qui sont géographiquement distribuées. Dans ce qui suit, nous présentons quelques projets qui font partie de différents domaines (voir tableau 1.1)

Tableau 1. 1 Quelques Projets de Grid

Projet	Description
SETI@home The Search for Extraterrestrial Intelligence **SETI@home [Set09]**	- Le projet a été initié par l'Université de Berkeley, son principe est d'exploiter la puissance inutilisée de millions d'ordinateurs connectés à Internet pour la recherche d'intelligence extraterrestre.
Genome @home **Genome@home [Gen09]**	- Ce projet étudie les interactions entre le génome et les protéines. - Près de 90 000 membres actifs proposent leurs machines personnelles pour ces calculs de recherche.
world community grid. La technologie, une réponse aux problèmes **World Community Grid [Wcg09]**	- C'est une communauté de calcul, qui a pour but de créer un lien entre des laboratoires de recherche qui ont besoin d'une énorme puissance de calcul, et des volontaires soucieux de participer à la recherche scientifique. - Permet à l'utilisateur de choisir sa participation sur un des projets lancés afin de lutter contre plusieurs maladies humaines
Rennes Sophia Lyon Grenoble Bordeaux Orsay Toulouse **Grid'5000 [Gri09]**	- L'objectif est de construire aux scientifiques un instrument pour la recherche dans le domaine de Grid. - Il est constitué d'une dizaine de centres équipés de fermes d'ordinateurs personnels.
Data GRID **Data Grid [Tdg09]**	- Projet financé par l'union européen, l'objectif est de fournir l'infrastructure de calcul de la prochaine génération. - Fournit le calcul intensif et l'analyse de larges bases de données qui sont partagées, à travers les communautés scientifiques distribuées.

1.4. ARCHITECTURE DE GRID

Nous présentons ici l'architecture de Grid en identifiant les principaux composants, leurs fonctionnalités et en indiquant comment se fait l'interaction entre eux.

Foster un des pères de Grid a donné une architecture protocolaire de Grid dans son article "*the anatomy of the grid*" **[FKT01]**. Elle est composée de cinq couches qui seront présentées dans la section suivante.

C'est une architecture extensible et standard qui répond aux besoins clés des organisations virtuelles. Cette architecture est basée sur un ensemble de protocoles, services, APIs et SDKs, et fondée sur les services et protocoles d'Internet. Elle est organisée en couches, ces dernières représentent un ensemble de fonctions comme le montre la figure 1.1 citée ci-dessous. Chaque couche fait appel à n'importe quelle fonction des couches inférieures.

L'architecture de Grid est basée sur le principe du "modèle en sablier" ("hourglass model" en anglais) ; le coeur du sablier définit un ensemble d'abstractions et de protocoles sur lequel différents comportements de haut niveau peuvent être construits (le haut du sablier), ces abstractions et protocoles peuvent être construits sur différentes technologies sous-jacentes (le bas du sablier). Le cœur du sablier est formé par les protocoles de la couche ressource et connectivité qui facilitent le partage des ressources individuelles.

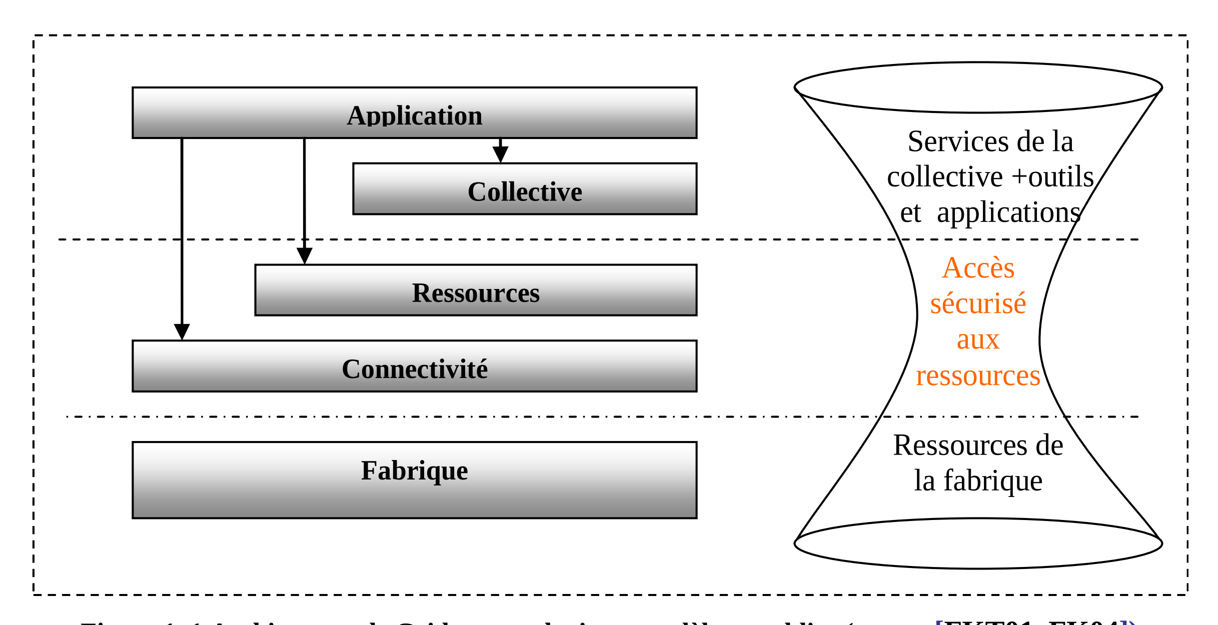


Figure 1. 1 Architecture de Grid par analogie au modèle en sablier (source [FKT01, FK04])

1.4.1. LA COUCHE FABRIQUE

Cette couche représente le niveau le plus proche des ressources physiques, elle regroupe l'ensemble des ressources partagées entre les différents participants. Une ressource peut être une

entité logique (systèmes de fichier distribués, ..) ou physique (ressources de calcul, ressources de stockage, le réseau, données, logiciels et capteurs spécialisés).

En pratique, toutes les ressources doivent au minimum implémenter des mécanismes qui permettent la découverte de leurs structure, capacités et état, et aussi des mécanismes de gestion des ressources qui permettent le contrôle de la qualité des service fournis, comme expliqué dans les exemples suivants :

- *Les* ressources *de calcul* : nous aurons besoin des mécanismes pour lancer les programmes, gestion et contrôle de leur exécution. les mécanismes d'introspection sont nécessaires pour la découverte des caractéristiques matérielles et logicielles de ces ressources, ils nous donnent aussi des informations sur l'état comme la charge courante.
- *Les* ressources *de stockage* : dans ce type de ressources il est nécessaire de déposer et récupérer des fichiers. Les ressources doivent pouvoir fournir des informations sur leur état comme l'espace disponible, il est nécessaire aussi de disposer des mécanismes qui permettent le contrôle des ressources utilisées pour des transferts particuliers de données.
- *Les* ressources *réseau* : des mécanismes de gestion ici sont nécessaires pour le contrôle à travers les ressources allouées pour des transferts réseau. Des mécanismes aussi sont nécessaires pour déterminer la charge et les caractéristiques du réseau.

1.4.2. LA COUCHE CONNECTIVITE

La couche connectivité définit les protocoles de communication et d'authentification nécessaires pour les transactions au sein de Grid.

- *les protocoles de communication :* permettent l'échange de données entre les diverses ressources de la fabrique. Ces protocoles sont basés sur les protocoles d'Internet ; IP pour le réseau, TCP et UDP pour le transport et DNS pour les applications.
- *Les protocoles d'authentification* : ils disposent des mécanismes sécurisés de vérification de l'identité des utilisateurs et des ressources, ces protocoles sont construits sur les services de communication. Les solutions d'authentification suivies ici disposent des caractéristiques suivantes :
  - *Single Sign On (SSO)* : l'utilisateur s'authentifie une seule fois pour pouvoir bénéficier des capacités offertes par le Grid.
  - *Délégation* : donner aux programmes le droit d'accéder aux ressources pour lesquelles l'utilisateur est autorisé d'accéder.

- *Intégration avec des solutions de sécurité locales* : chaque site ou fournisseur de ressources peut utiliser ses propres politiques de sécurité et le Grid doit pouvoir interopérer avec ces différentes solutions locales.
- *Relations de confiance* : un utilisateur voulant utiliser plusieurs ressources appartenant à différents fournisseurs n'aura pas besoin de l'intervention des fournisseurs pour la configuration de l'environnement de sécurité.

1.4.3. LA COUCHE RESSOURCE

La couche ressource fournit un ensemble de protocoles (APIs et SDKs) pour négocier l'accès aux ressources individuelles, l'initiation, la surveillance, le contrôle et la gestion d'une éventuelle facturation quant à leur utilisation. Elle est basée sur les protocoles de communication et d'authentification de la couche connectivité et fait appel aux différentes fonctions offertes par la couche fabrique afin d'accéder aux ressources locales et de les contrôler.

Deux types de protocoles peuvent être distingués :

- **Protocoles d'informations**, utilisés pour obtenir des informations sur la structure et l'état d'une ressource, ces informations peuvent être statiques (la configuration, la politique de sécurité, etc.) ou dynamiques (exemple, la charge courante).
- **Protocoles de gestion,** utilisés pour négocier l'accès à une ressource partagée en spécifiant par exemple les opérations à effectuer (accès au données, création d'un processus,..) et les critères sur les ressources à partager (qualité de service, supporter une réservation avancée).

1.4.4. LA COUCHE COLLECTIVE

Permet de coordonner des ressources multiples. Contrairement à la couche ressource qui est basée sur les interactions avec une seule ressource, la couche collective contient des protocoles et des services (et des APIs et SDKs) qui ne sont pas associés pour une ressource spécifique mais qui sont globales de nature et capturent les interactions entre une collection de ressources.

Les composants de la couche collective implémentent une variété de comportements partagés sans placer de nouveaux critères sur les ressources à partager, plusieurs services sont offerts par cette couche, citons les suivants :

- La découverte des ressources d'une OV et leurs propriétés à l'aide du *service d'annuaire.*

- La co-allocation des ressources par le *service de co-allocation* et le *service de brokering* appelé *courtier de ressource* qui se charge de chercher dans les annuaires les meilleures ressources qui conforment le plus aux caractéristiques requises par les clients.
- l'ordonnancement des tâches s'exécutant sur des ressources spécifiques par le *service d'ordonnancement.*
- La réplication des données par *le service de réplication.*
- L'échange coordonné d'informations synchrone ou asynchrone entre un grand nombre d'utilisateurs appartenant à différentes communautés à l'aide des services collaboratoires.

Les composants de la couche collective peuvent être dédiés pour les besoins des utilisateurs d'une communauté spécifique, OV ou un domaine d'application particulier, par exemple le service de co-allocation est spécifique à un ensemble particulier de ressources réseau. D'autres composants sont plus généraux comme le service de réplication qui gère une collection de systèmes de stockage pour de multiples communautés.

1.4.5. LA COUCHE APPLICATION

Finalement, les applications utilisateurs forment la dernière couche du modèle, ces applications sont construites et en appelant les services définis dans les autres couches. Dans chaque couche il existe un certain nombre de protocoles et d'APIs pour pouvoir accéder aux services (gestion de ressources, accès aux données, etc.).

Elle contient des outils qui peuvent aider les développeurs pour écrire des applications adaptées au Grid et aider les utilisateurs à déployer et exécuter leurs applications. Deux types d'outils peuvent être distingués ; les modèles de programmation Grid et les environnements d'exécution des applications Grid. Les modèles de programmation Grid fournissent des abstractions de haut niveau qui seront utilisées pour écrire des applications. Par contre, les environnements d'exécution des applications Grid sont des outils logiciels intégrés sur lesquels les utilisateurs peuvent déposer et exécuter leurs applications et auront l'impression d'une exécution bureau (locale) pendant que l'outil déploie l'application sur les ressources de Grid automatiquement et de manière transparente **[BCDS04]**.

1.5. L'ARCHITECTURE OGSA

1.5.1. RAISONS D'APPARITION

Le Grid est un système qui rassemble un ensemble de ressources d'ordre matériel et logiciel, distribuées et hétérogènes, à cause de cette héterogenieté le Global Grid Forum (GGF) **[Ogf09]** a commencé la conception de standards pour le Grid.

Globus Toolkit 2 (GT2) **[Tga09]** et ses versions précédentes ont été développés pour construire les systèmes Grid. Cependant, les systèmes Grid basés sur Globus à ce stade là, sont hétérogènes car ils sont développés par des protocoles hétérogènes ce qui rend l'interopérabilité un aspect presque impossible. Dans cette période, une nouvelle infrastructure est développée en parallèle qui permet le développement et l'intégration d'applications dans des environnements hétérogènes ; c'est celle des Web services **[Wsa09]**.

Les concepteurs ont commencé petit à petit à combiner les deux technologies Grid et Web services afin de résoudre les problèmes d'héterogenieté, et OGSA (Open Grid Services Architecture) donc a été proposée pour construire de manière standard les systèmes Grid orientés services.

OGSA introduit le concept des *Grid services*, qui sont des Web services avec quelques extensions pour répondre aux besoins spécifiques de Grid. Par la suite, OGSI (Open Grid Services Infrastructure) qui est une spécification technique pour l'implémentation des Grid services est utilisée par la nouvelle version de Globus Toolkit (GT3) pour implémenter OGSA dans le contexte des Web services. Les Web services standards sont persistants et sans état, mais les Grid services définis avec OGSI peuvent être transitoires et à état, cependant la spécification OGSI a été lourde et n'a pas pu s'adapter correctement avec les Web services existants et l'outillage XML. En janvier 2004, Globus Alliance et IBM en conjonction avec HP ont introduit WSRF (Web Service Resource Framework) pour résoudre ce problème. La figure 1.2 montre l'évolution des deux technologies et enfin, leur convergence.

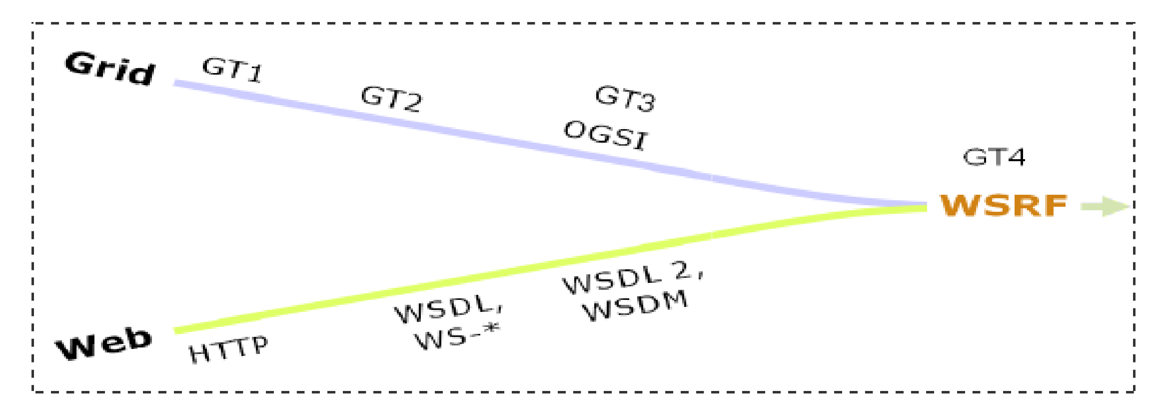


Figure 1. 2 Convergence des technologies Grid et Web services

1.5.2. PRESENTATION DE L'ARCHITECTURE OGSA

Dans cette section nous présentons l'architecture OGSA, en montrant le rôle des différentes technologies et standards dans la construction de cette architecture.

Lors de la description des organisations virtuelles deux notions semblaient importantes **[FKT01]** ; les ressources physiques à partager et les services offerts par ces ressources. Dans OGSA toutes les ressources (calcul, stockage, réseaux, données, etc.) sont modélisées comme des services **[FKNT02]**, par conséquent, OGSA a adopté l'architecture orientée service en standardisant les interfaces des services de Grid et les protocoles utilisés pour invoquer ces particulières interfaces.

OGSA a été proposée comme une infrastructure pour les systèmes et les applications qui nécessitent l'intégration et la gestion des services dans les organisations virtuelles dynamiques, hétérogènes et distribuées **[FKNT02]**.

OGSA est une architecture commune et standard pour les applications à base de Grid dans le but de standardiser tous les services qui peuvent exister dans une application basée Grid (services de gestion des tâches, services de gestion de ressources, services de sécurité, etc.). L'objectif de cette standardisation est la mise en œuvre de l'interopérabilité **[FKNT02]**, *qui est un besoin critique dans les environnements Grid distribués* **[FKT01]**.

L'architecture OGSA est organisée en couches (voir figure 1.3), chaque couche a ses propres fonctionnalités. Les couches seront présentées séparément dans les sections qui suivent.

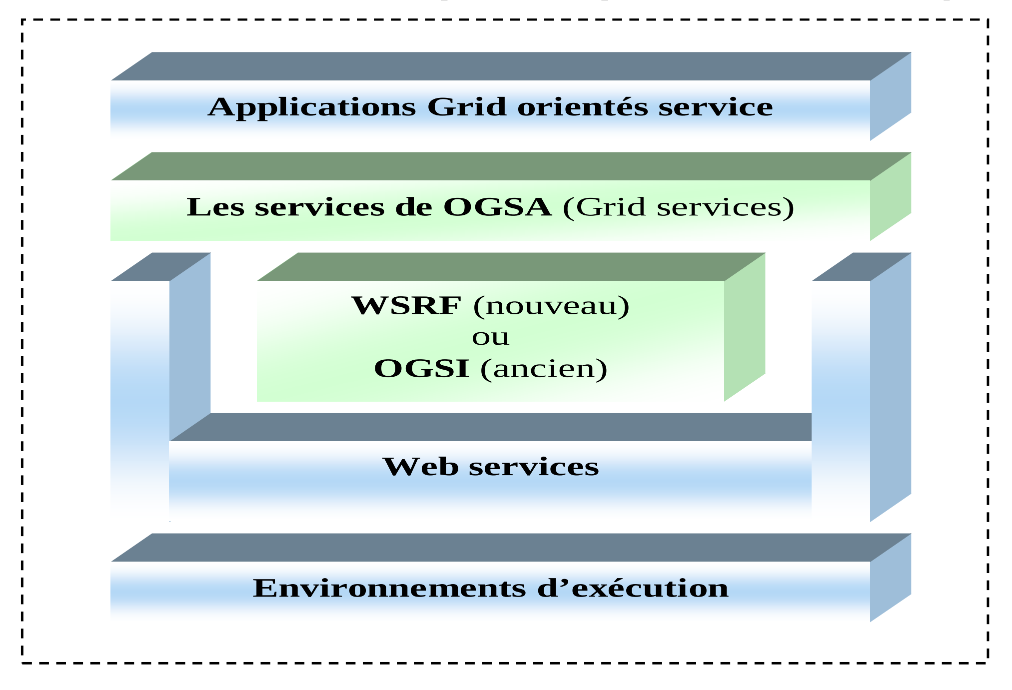


Figure 1. 3 Architecture OGSA (source [LB05])

1.5.3. LES GRID SERVICES

1.5.3.1. Description et définition

OGSA définit ce qu'on appelle *le Grid service qui est défins ci-dessous :*

Le Grid service est un Web service qui fournit des interfaces bien définies tout en respectant des conventions spécifiques **[FKNT02]**.

Les interfaces et les conventions qu'un Grid service définit adressent la création, la découverte et le nommage d'un service, la gestion de sa durée de vie, la notification et les protocoles de communications choisis. Ces interfaces et conventions sont liées à la gestion des instances de services transitoires.

Dans une OV, il existe deux types de services ; les *services persistants statiques* qui traitent et manipulent des tâches complexes invoquées et soumises par le client, et les *instances de services transitoires* pour gérer les interactions avec état des activités particulières. Les instances de services sont des entités légères crées dynamiquement pour gérer des activités à courte durée. Par exemple, requête à une base de données, opération de datamining, l'allocation de bande passante réseau. Le Grid service définit plusieurs interfaces qui sont bien détaillées dans **[FKNT02]** et résumées dans le Tableau 1.2 ci-dessous:

Tableau 1. 2 Les interfaces de Grid service définies par OGSA (source [FKNT02])

portType	Opération	Description
GridService	FindServiceData	Trouver des informations concernant l'instance de Grid service.
	SetTerminationTime	Indiquer le temps de terminaison d'une instance d'un Grid service.
	Destroy	détruire une instance d'un Grid service.
NotificationSource	SubscribeTo-NotificationTopic	S'inscrire pour recevoir des notifications lorsque certains évènements se produisent au niveau d'un Grid service.
NotificationSink	DeliverNotification	Notifier de la survenue d'un événement.
Registry	RegisterService	Enregistrer un service auprès de l'annuaire.
	UnregisterService	Enlever un service de l'annuaire.
Factory	CreateService	Créer une nouvelle instance d'un Grid service.
HandleMap	FindByHandle	Obtenir une référence sur un service.

En voici quelques exemples :

- *Création dynamique d'un service* : La création dynamique des instances d'un service se fait dans OGSA par une interface standard (factory).
- Gestion *du cycle de vie* : OGSA est dotée de mécanismes produisant des services qui permettent la récupération d'état et services associés aux opérations échouées, cela se fait par deux opérations standards : Destory et SetTerminationTime.
- *La* notification : Une collection de services dynamiques distribués doit être capable de notifier les uns les autres de façon asynchrone des changements importants de leurs états.

1.5.3.2. Caractéristiques d'un Grid service et processus d'invocation

Cette section présente les caractéristiques de base d'un Grid service telles qu'elles sont définies par OGSA, le schéma ci-dessous (figure 1.4) résume le processus d'invocation d'un Grid service ainsi que les éléments et les opérations intervenantes :

L'existence de l'état permet de distinguer une instance d'un service d'une autre qui fournit la même interface. Le terme *instance* de Grid service est utilisé pour référer à une instanciation particulière de Grid service. Cette instance est crée par une autre instance de service appelée *Grid service factory*. Les instances de Grid service peuvent être crées et détruites dynamiquement.

Dans OGSA les instances des Grid services sont crées avec une durée de vie spécifique, dès que cette durée expire l'instance est détruite dynamiquement avec une libération des ressources associées, mais cette durée peut être étendue par une demande explicite de l'utilisateur ou de Grid service. Chaque instance de Grid service est identifiée par le GSH (Grid Service Handle) et le GSR (Grid Service Reference). Le GSH aide les utilisateurs à localiser un service, mais ne leur donne pas des informations suffisantes pour communiquer directement avec l'instance de ce service, c'est le rôle du GSR qui est un document en WSDL contenant les informations nécessaires pour l'utilisateur pour la communication avec le service. Chaque GSR a une durée de vie limitée, cette durée peut varier durant le cycle de vie du service. Lors de la création de nouvelles instances par l'opération *CreateService* du service Factory le GSH et le GSR initiaux de cette instance sont retournés.

Une instance de Grid service implémente une ou plusieurs interfaces, chaque interface définit un ensemble d'opérations qui sont invoquées par l'échange d'une suite de messages. OGSA introduit deux caractéristiques majeures :

- Management de l'état de service : les Grid services peuvent être soit avec ou sans état.

- Management de la durée de vie de service : les Grid services peuvent être soit persistants ou transitoires.

Le processus d'invocation de Grid service est illustré dans le schéma de la figure 1.4 ci-dessous.

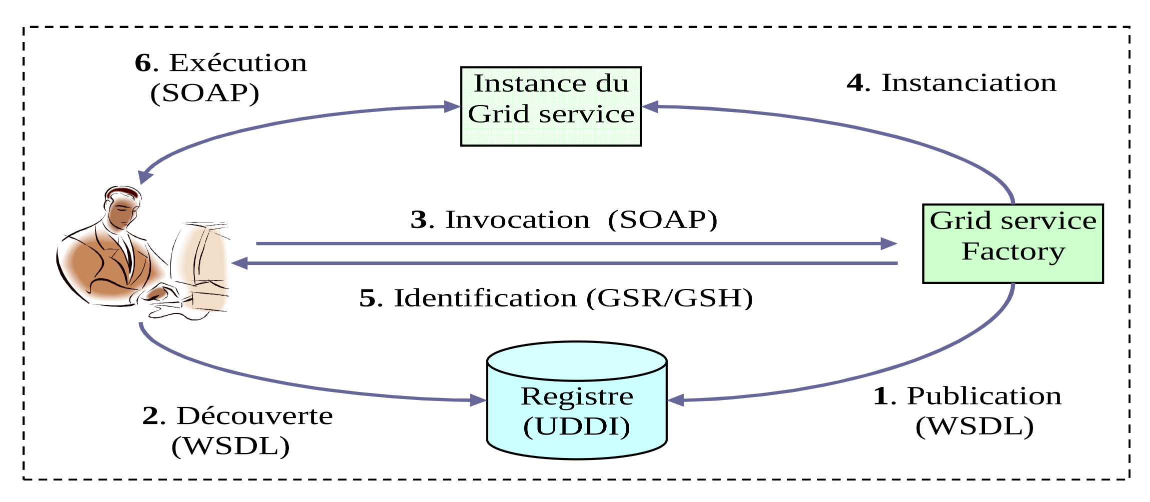


Figure 1. 4 Processus d'invocation d'un Grid service

1.5.3.3. Exemples de Grid services

La couche qui présente les fonctionnalités et les services OGSA est celle des Grid services. Nous avons définit qu'est ce qu'un Grid service et les interfaces qu'il présente, maintenant nous citons de manière brève quelques exemples sur les fonctionnalités offertes par ces services sachant que les services que nous allons citer sont bien détaillés dans le chapitre suivant "Globus Toolkit 4" .

- Services de gestion des tâches : permet la gestion des jobs soumis par les utilisateurs au Grid et de suivre leurs exécutions.
- Services de sécurité : sécuriser le fonctionnement de Grid. Plusieurs services sont offerts tel que, l'authentification, l'autorisation et la confidentialités des échanges.
- Services d'information : ce service donne des informations sur l'état et la structure de Grid, à partir de ces informations-là les applications peuvent découvrir les ressources qui répondent à leurs besoins.

Les services définis par OGSA travaillent en coopération, par exemple le service de gestion des tâches doit consulter d'abord le service d'information pour trouver les ressources nécessaires pour l'exécution des jobs.

1.5.4. LES WEB SERVICES

Il est nécessaire de donner une idée sur les Web services pour bien comprendre le rôle des standards utilisés pour former une solution à OGSA, cette dernière est basée sur la technologie des Web services mais avec quelques extensions en introduisant des interfaces spécifiques et des conventions, cette extension apparaît dans les trois points suivants **[FKNT02]** :

- Les services d'un environnement Grid sont caractérisés par leur nature dynamique et transitoire, chaque service est instancié lors de son utilisation, et la durée de vie d'une instance varie d'un service à un autre. Par conséquent, les Grid services ont besoin d'interfaces pour gérer leur création, destruction et cycle de vie.
- Les Grid services peuvent avoir des données et des attributs qui leur sont associés, on dit donc qu'ils possèdent un état, par conséquent les Web services doivent supporter les données d'état associées aux Grid services.
- Les services doivent notifier les clients des changements qu'ils subissent.

1.5.4.1. Définitions

Les Web services est une autre technologie de calcul distribué (comme CORBA, RMI, EJB, etc.), ils permettent de créer des applications *client/serveur* et permettent aussi aux différentes applications provenant de différentes sources de communiquer les unes avec les autres **[Bor05]**.

L'objectif des Web Services est de faciliter l'accès aux applications entre entreprises et ainsi de simplifier les échanges de données. Ils poursuivent un vieux rêve de l'informatique distribuée où les applications pourraient interopérer à travers le réseau indépendamment de leur plate-forme et de leur langage d'implémentation. Dans ce sens, ils s'inscrivent dans la continuité d'initiatives telles que CORBA (Common Object Request Broker Architecture, de l'OMG) en apportant toutefois une réponse plus simple, s'appuyant sur des technologies et standards reconnus et maintenant acceptés de tous.

La technologie des Web services est initiée par IBM et Microsoft, puis en partie normalisée sous l'égide du W3C, elle est maintenant acceptée par l'ensemble des acteurs de l'industrie informatique sans exception. C'est surtout ce point qui fait des Web Services une technologie révolutionnaire **[Web09]**. Le World Wide Web Consortium (W3C) définit un Web service ainsi:

Il s'agit d'une technologie permettant à des applications de dialoguer à distance via Internet, et ceci indépendamment des plates-formes et des langages sur lesquelles elles reposent. Pour ce faire, les Web services s'appuient sur un ensemble de protocoles standardisant les modes d'invocation mutuels de composants applicatifs **[Sew09]**.

Lorsque on parle des Web services on a toujours besoin de montrer le rôle principale des nouveaux standards dans la définition de cette nouvelle technologie. En utilisant les standards Internet existants *TCP*/*IP* et *HTTP*, les Web services commencent par le *XML* un standard qui établit un format pour l'échange de données et les fonctionnalités d'une application. Ce standard a conduit à la naissance de nouveaux standards et protocoles comme le *SOAP* (l'établissement d'un format commun pour l'adressage des messages), *WSDL* (description standard des fonctionnalités d'un Web service et des instructions sur la façon d'y accéder) et le *UDDI* (produit une façon uniforme pour enregistrer les Web services afin de faciliter leurs recherche et découverte par d'autres) **[HB02].** L'architecture des Web services s'est imposée grâce à sa simplicité, à sa lisibilité et à ses fondations normalisées. Les Web services s'articulent autours des trois acronymes suivants :

SOAP (Simple Object Access Protocol)

C'est un protocole d'échange inter-application basé XML indépendant de toute plate-forme, utilisé pour l'invocation des services Web et définit la structure des messages échangés par les applications.

- Le mécanisme de transport des messages *SOAP* peut être le protocole *HTTP*, *SMTP*, *FTP*, etc.
- *SOAP* utilise les mécanismes *HTTP* et *XML*. *HTTP* comme un mécanisme d'invocation de méthodes. *XML* pour structurer les requêtes et les réponses, indiquer les paramètres des méthodes, les valeurs de retours, et les éventuelles erreurs de traitements.

Un message *SOAP* est un document *XML* qui possède une enveloppe *SOAP* et éventuellement une déclaration *XML* **[Wsa03]**.

WSDL (Web Service Description Language)

Basé sur *XML*, permet de décrire le service Web, il décrit une interface publique d'accès à un service et décrit comment accéder à ces services et les opérations qu'ils exposent. Il s'agit donc d'un langage qui standardise les schémas *XML* utilisés pour établir une connexion entre émetteurs et récepteurs. Un *document WSDL* est composé de deux parties :

- Une partie abstraite qui contient les éléments suivants :
  - **Types** : définitions des types de données échangées.
  - **Message** : définition abstraite des données transmises entre le client et le serveur lors d'une opération sur le port d'un service.
  - **Type de port (Portype):** ensemble d'opérations correspondant chacune à un message entrant ou sortant.

- La partie concrète spécifie :
  - **Rattachement (Binding)** : protocole de communication et format des données échangées pour un port. WSDL offre deux extensions de rattachement prédéfinies : SOAP et HTTP.
  - **Port (Endpoint)**: adresse (assure l'unicité du rattachement), il associe une adresse réseau avec un rattachement.
  - **Service** : regroupe un ensemble de ports qui implémentent une interface commune. Il définit ou un service particulier peut être contacté. Il relie ensemble le nom de service, le binding et l'adresse **[War05]**.

UDDI (Universal Description Discovery and Integration)

Permet la publication d'un service dans un registre des services, l'identification et la découverte des fournisseurs de ces services. UDDI se comporte lui-même comme un Web service dont les méthodes sont appelées via le protocole SOAP. L'annuaire UDDI est consultable de différentes manières :

- Les **pages blanches,** comprennent la liste des entreprises ou les organisations ayant publié des informations dans le répertoire ainsi que des informations associées à ces dernières. Nous y retrouvons donc des informations comme le nom de l'entreprise, ses coordonnées, la description de l'entreprise mais également l'ensemble de ses identifiants.

- Les **pages jaunes,** recensent les Web services de chacune des entreprises sous le standard WSDL. Elles décrivent de manière non technique les services proposés par les différentes organisations

- Les **pages vertes,** fournissent des informations techniques précises sur les services fournis. Ces informations concernent les descriptions de services ou encore les processus métiers associés.

Grâce à cette triple lecture, l'ensemble des informations utiles sont accessibles **[Vli04]**.

1.5.4.2. L'invocation d'un Web service

Un Web service est crée afin d'être utilisé par un grand nombre d'utilisateurs, donc chaque organisation publie ses services dans un registre d'annuaire appelé UDDI afin qu'il puisse être découvert par un client voulant l'utiliser. La spécification UDDI est utilisée pour publier les services et permettre leur découverte. Les étapes nécessaires pour invoquer un Web service sont résumées ainsi et schématisées dans la figure 1.5 :

1) la découverte d'un Web service, car le client peut ne pas avoir des connaissances au préalable sur le Web service qu'il veut invoquer. Le client envoie une requête à l'annuaire pour la recherche d'un Web service.

2) Le service de découverte va répondre en informant le client des serveurs qui peuvent fournir le service recherché.

3) Maintenant, l'emplacement du Web service est connu par le client, mais ce dernier n'a aucune idée sur comment l'invoquer. Il envoie alors une requête au serveur

4) Le Web service répond par le langage WSDL en indiquant comment invoquer le Web service.

5) Une fois le client connaît ou se trouve le service et comment l'invoquer, l'invocation est faite à l'aide du langage SOAP ; le client envoie un document XML représentant sa requête.

6) Le Web service va répondre avec une réponse SOAP ou un message d'erreur. Il exécute la requête et retourne le résultat.

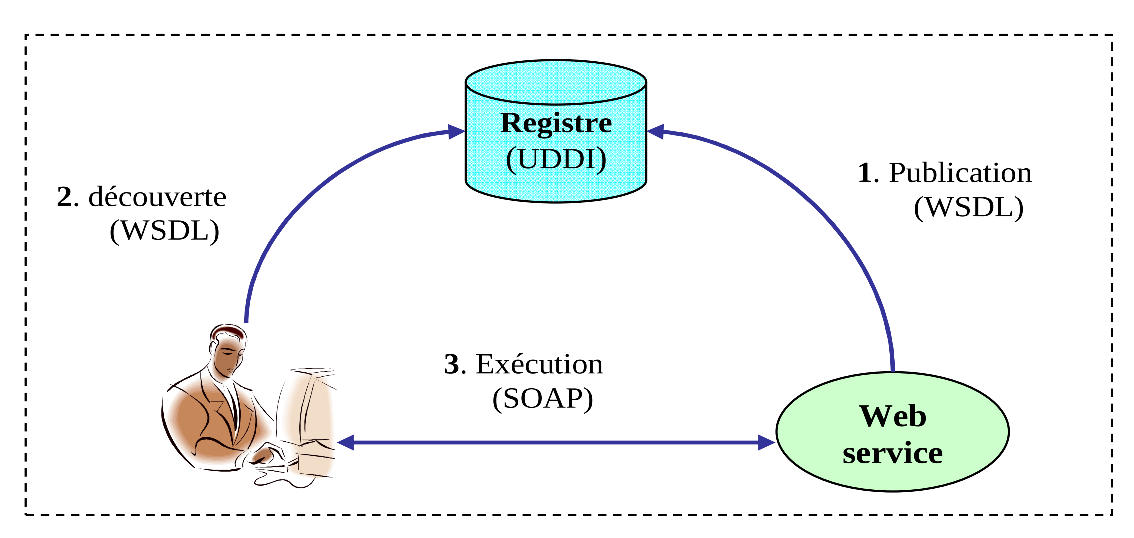


Figure 1. 5 Le cycle de vie d'un Web service

1.5.4.3. Avantages et inconvénients des Web services

Les Web services présentent plusieurs avantages mais aussi ont quelques inconvénients, le tableau 3 ci-dessous représente les principaux avantages et inconvénients des Web services tirés à partir de **[Bor05, Jon06]**.

Tableau 1. 3 Avantages et inconvénients des Web services

Avantages	Inconvénients
- Simplifient la virtualisation (l'encapsulation de plusieurs implémentations derrière une interface commune) - Indépendance des plateformes et langages de programmation (utilisation du langage XML) - Dédiés aux applications faiblement couplées (Contrairement aux autres technologies comme EJB, CORBA où le client et le serveur sont très dépendants). - Bien adaptées aux applications orientés Grid. - facilitent l'intégration des composants hétérogènes.	- Comportement orienté objet. - Orientation client/serveur. - Pas d'adaptation utilisateur - Pas de mémoire (sans état) - Pas de gestion de cycle de vie. Pas de manipulation de la conversation (interaction de type requête/réponse). - Communication synchrone. - Ils sont passifs et ne prennent pas en compte l'autonomie des composants

1.5.5. LE ROLE DE OGSI ET WSRF

Il existe deux standards principaux pour la standardisation des Grids, qui sont OGSI (Open Grid Service Infrastructure) et WSRF (Web Service Resource Framework). Dans cette section nous décrivons de manière détaillée chacun de ces standards.

1.5.5.1. OGSI

Le standard OGSI est le composant de base de l'architecture OGSA, c'est une infrastructure logicielle pour le Grid basée sur les standards des Web services. L'objectif de OGSI est de permettre l'interopérabilité entre les composants logiciels de OGSA **[JEF04]**. La spécification OGSI définit une instance d'un Grid service comme un Web service qui se conforme à un ensemble de conventions ; ces conventions sont exprimées par WSDL sous formes d'interfaces, des extensions et des comportements d'un service. Le Grid service doit être doté de mécanismes qui lui permettent le contrôle et la gestion de son état **[JEF04]**. La figure 1.6 montre les composants de base de OGSI. Les points essentiels qui caractérisent son architecture sont résumés ci-dessous :

- L'extension de WSDL pour fournir des mécanismes de description des données d'état.
- Les interfaces et les comportements qui décrivent et maintiennent le cycle de vie d'un service (création et gestion de services, gestion des références d'une instance et des collections, notification des changements d'état).

- L'interopérabilité au niveau message est un concept clé de ce standard, et elle est basée sur XML pour définir les formats de messages.

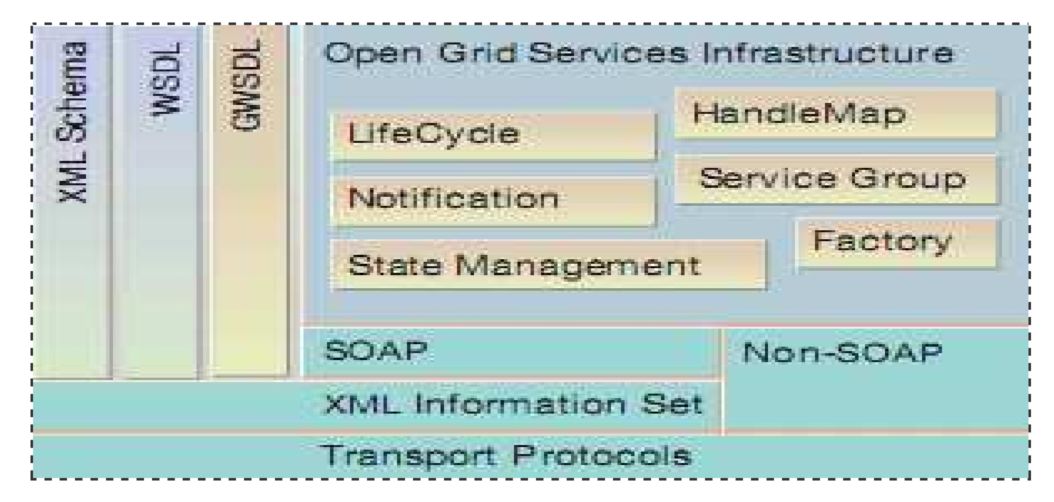


Figure 1. 6 Les composants de OGSI (source [JEF04])

OGSI est concernée par la création, l'adressage, l'inspection et la gestion de la durée de vie des Grid services à état **[FKNT02]**. Elle utilise les schémas XML et WSDL pour introduire l'ensemble des concepts définis dans OGSA :instances des Web services à état, données et méta données de services, notification asynchrone des changements d'état, références des instances de services, collections d'instances, déclaration et inspection des données d'état d'un service **[FCF+05]**.

L'évolution des Web services a une grande influence sur l'évolution de OGSI ; de nouvelles spécifications ont émergé et simplifié les idées exprimées dans OGSI. Par exemple, les spécifications WS-addressing **[BCC+04]** et WS-metadataexchange. La communauté des Web services a donné des critiques sur la spécification OGSI **[CFF+04a] :**

- Plusieurs objets dans une seule spécification.
- Problèmes rencontrés avec l'outillage des Web services et XML.
- OGSI est purement orienté objet. OGSI V1.0 modélise les ressources à état comme un Web service qui encapsule l'état de la ressource, l'identité et la durée de vie du service.
- L'introduction de WSDL 2.0. cependant, les capacités de OGSI sont mieux exprimées avec WSDL 1.1.

1.5.5.2. WSRF

Le Web Service Resource framework (WSRF) ; un effort entre les communautés de Grid et Web services, est un ensemble de spécifications des Web services. Il introduit le WS-Resource

pour modéliser et gérer les informations d'état dans le contexte des Web services qui sont souvent sans état. Il spécifie comment rendre les Web services à état.

Un Web service est dit à état s'il peut garder l'information d'une invocation à une autre. La notion d'état n'est pas nécessaire pour toutes les applications, par contre les applications Grid ont en besoin **[Bor05]**. L'idée est de donner la possibilité au Web service de garder des informations d'état dans une entité autre que le service même, donc la solution est de sauvegarder l'état dans une entité séparée appelée 'ressource', et à chaque fois qu'un client nécessite une interaction à état avec le Web service il suffit de lui indiquer simplement la ressource à utiliser.

WS-Resource est défini comme la composition d'un Web service et une ressource à état qui est exprimée comme une association d'un document XML avec le porType d'un Web service **[CFF+04b]**. Chaque ressource a un identificateur unique qui est encapsulé dans 'reference endpoint', et qui est utilisé par un client pour déterminer la ressource à état qui sera utilisée pour l'exécution du Web service en question. De ce fait, l'adresse d'une WS-Resource particulière est nommée 'endpoint reference'.

WSRF est un ensemble spécifications, comme montré dans la figure 1.7, qui servent à la gestion de WS-resources et qui sont décrites ci-dessous.

Figure 1. 7 WSRF avec les spécifications des Web services (Source [JEF04])

- *WS-ResourceLifetime* : C'est l'ensemble des mécanismes nécessaires pour la destruction des WS-Resource à travers les messages échangés entre le client et le Web service, immédiatement ou après un certain temps. Cette spécification adresse trois aspects importants du cycle de vie de WS-Resource à savoir la création, l'identité et la destruction. La création d'une WS-Resource se fait à l'aide de WS-Resource factory, le résultat de

l'opération de création est sous forme d'un message contenant au moins un '*endpoint reference'* qui fait référence à la nouvelle WS-Resource créee.

- *WS-ResourceProperties* : Définit le type et les valeurs contenus dans l'état de WS-Resource, ces informations peuvent être consultées et modifiées par les clients à travers l'interface du Web service invoqué.
- *WS-BaseFaults* : Cette spécification permet de donner une façon standard pour signaler les erreurs qui peuvent survenir lors de l'invocation d'un Web service. L'utilisation des messages d'erreurs communs a facilité la détermination du problème et la gestion de l'erreur.
- *WS-ServiceGroup* : Définit les outils par lesquels les Web services et les WS-Resources peuvent être groupés ensemble. Cette spécification peut être utilisée par exemple, pour organiser des collections de WS-Resources afin de construire des registres, ou des services qui peuvent effectuer des opérations collectives sur une collection de WS-Resources.
- *WS-RenewableReferences* : Cette spécification définit des mécanismes standards pour permettre aux Web services de renouveler les '*endpoint references'* (des WS-Resources) lorsque une référence devient invalide. Elle remplace le concept de GSH et GSR dans la spécification OGSI 1.0. WS-RenewableReferences devrait fournir les informations nécessaires sur les '*endpoint reference*' pour extraire une nouvelle '*endpoint reference'* dans le cas où la référence devient invalide.

D'autres spécifications :

- *WS-Notification* : C'est une spécification qui ne fait pas partie de WSRF mais qui est liée à ce dernier, elle permet au Web service de notifier les clients des changements qui peuvent être apportés à un Web service ou à ses ressources.
- *WS-Addressing* : fournit des mécanismes pour adresser les Web services. La spécification *WS-Addressing* définit des éléments XML pour identifier les 'endpoints' des Web services et pour inclure l'identification du 'endpoint' dans les messages.

1.6. INTERGICIELS POUR LE GRID

Plusieurs intergiciels ont été proposés pour implémenter le Grid, chaque intergiciel a ses propres objectifs et son mode d'utilisation. Dans ce qui suit, nous présentons les intergiciels les plus utilisés, notamment UNICORE, BEOWULF, LEGION, CONDOR, et GLOBUS. Nous avons choisi ces intergiciels en raison de leur importance et leur maturité.

1.6.1. UNICORE

Est un projet européen dont l'objectif est la fédération de différents centres de calcul en Europe. UNICORE possède des fonctionnalités similaires à celle de GLOBUS (qui sera présenté par la suite) ; sécurité, gestion des ressources, système d'information et gestion des fichiers mais propose une interface d'utilisation conviviale qui intègre des fonctionnalités de haut niveau.

1.6.2. BEOWULF

Développé par la NASA, ne sert qu'a constituer des grappes de machines dans un réseau local et privé. A partir de BEOWULF, nous ne pouvons fabriquer que des machines multi processeurs à mémoire distribuée.

1.6.3. LEGION

Développé par l'université de Virginie au USA, il est d'une conception orientée objet. Il est très ouvert sur le monde extérieur mais ne gère qu'un ensemble restreint d'ordonnanceurs. Il a pour objectif d'offrir une vision unique des ressources de Grid, pour cela il propose un modèle objet où tout élément de Grid (ressources de calcul, applications, fichiers, …) est représenté par un objet.

1.6.4. CONDOR

Est un des premiers intergiciels intégrés qui est né à la fin des années 1980. Condor vise les architectures de type réseau de stations de travail et a pour objectif de récupérer les cycles inutilisés des noeuds de calcul. Les stations de travail forment un Condor Pool, et un gestionnaire central est chargé de distribuer les tâches sur les stations. Les utilisateurs peuvent spécifier la nature des ressources requises pour leurs tâches grâce au système Class Add (Classified Advertisements).

1.6.5. GLOBUS

Est le logiciel qui couvre le plus de technologies et de protocoles dans le monde du calcul distribué, développé en collaboration entre des universités et laboratoires américains.

Parmi cet ensemble d'intergiciels et d'autres qui n'ont pas été présentés ici nous avons choisi Globus et cela est du à plusieurs raisons :

- Il est en open source,
- Contient des services avancés utilisables pour construire des applications Grid,

- Il n’est pas dédié pour un type particulier d’application,
- Implémente la plupart des services définis par OGSA,
- …

Nous avons pu l’installer et le tester et il présente un bon outil pour les applications Grid. Malgré sa complexité et la difficulté rencontrée dans le processus d’installation, il reste le logiciel qui répond le plus aux besoins des applications basées Grid. Les détails de Globus seront présentés dans le chapitre suivant.

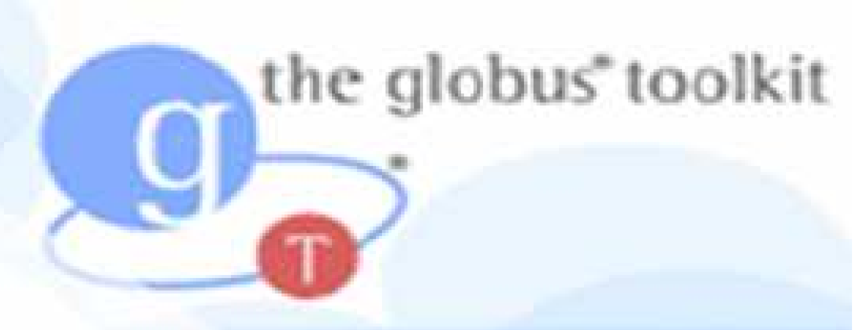
1.7. CONCLUSION

Dans ce chapitre, nous avons présenté le Grid, ses objectifs, son principe, architecture, standards, domaines d’application et ses intergiciels. En effet, pour ce faire, nous avons adopté une approche descendante dans la présentation des différentes parties de ce chapitre ; en commençant d’abord par des généralités pour aller ensuite à présenter les détails. L’enchaînement des titres choisi dans ce chapitre, et particulièrement la manière dont les différentes parties sont présentées est du en premier lieu à la complexité du domaine et sa richesse ; pour pouvoir comprendre qu’est ce que le Grid il faut connaître d’abord un large nombre de concepts.

Nous avons notamment tenté de répondre aux questions qui peuvent être posées : qui va utiliser le Grid ? Quels sont ses composants de base? Quels sont les domaines d’application ? Quels sont les standards utilisés ? Finalement, nous avons présenté brièvement quelques intergiciels de Grid.

Pour conclure, la raison principale de Grid est de répondre aux besoins des organisations virtuelles en leur offrant une infrastructure qui permet le partage coordonné de ressources distribuées et hétérogènes faisant partie de différentes organisations et institutions. Pour un besoin de standardisation et pour mettre en œuvre l’interopérabilité entre les différentes entités de Grid l’architecture OGSA a été proposée qui permet de standardiser les différents services offerts par le Grid en définissant des interfaces standards avec un accès standard aussi à ces interfaces. Pour ce faire, les technologies Grid et Web services ont été combinées.

Le domaine d’application de Grid est devenu très large (scientifique, industriel, commercial, ..) et la collaboration reste un des principes de base suivi par ses différentes modalités d’usage, ce qui donnera aux individus, institutions et organisations de différents niveaux, compétences et moyens la possibilité de s’améliorer car l’union fait la force !



CHAPITRE 2

L'INTERGICIEL GLOBUS

Globus Toolkit, est devenu une architecture de référence. Ses technologies de base décrivent au mieux les éléments indispensables au fonctionnement de base de Grid [Glo3].

Globus Toolkit est un logiciel qui implémente le Grid, développé par l'alliance Globus qui est une communauté d'organisations et d'individus qui collaborent ensemble pour développer une solution pour les technologies de Grid, à savoir le partage sécurisé de ressources de calcul, de bases de données et d'instruments spécifiques pour des organisations virtuelles dynamiques hétérogènes et distribuées.

Globus Toolkit (GT) a été développé depuis 1990, et il a évolué avec le temps en fonction des objectifs visés et les technologies utilisées ; GT1, GT2, GT3, et on est aujourd'hui à la version 4. Dans ce chapitre nous nous intéressons à GT4 qui est l'implémentation de l'architecture de services Grid OGSA ; résultat de la convergence de la technologie de Grid et celle des Web services.

Le but de ce chapitre est de décrire les principaux composants de Globus Toolkit 4 afin de donner une idée sur les services qu'il offre et comment ces derniers interagissent entre eux.

2.1. INTRODUCTION

GT4 est un logiciel en open source, il contient différents services de haut niveau pour la construction des systèmes de Grid, une grande partie de GT4 est l'implémentation de WSRF sur lequel la plupart des services sont basés.

GT4 est une collection de solutions pour la construction des applications collaboratives et distribuées. Le besoin d'assurer la collaboration scientifique et le partage de ressources pour résoudre des problèmes complexes était la motivation principale pour le développement de la première version de Globus Toolkit[3]. Vers 2002, GT2 est apparu, mais ses composants ne sont pas basés sur les Web services. La version 3 (GT3) inclut l'implémentation de OGSI, elle a offert de nouveaux services comme le RFT et le RLS qui seront présentés dans les sections suivantes de ce chapitre, et la majorité de ses composants sont basés sur les Web services. Finalement la version 4 (GT4), c'est la mise en œuvre de l'architecture OGSA dont l'objectif est l'utilisation d'interfaces et de protocoles standards pour mettre en œuvre l'interopérabilité entre les ressources de Grid, elle est réalisée en avril 2005 et c'est la convergence et la fusion des Web services et Grid services. Cette version offre de nouveaux services par rapport à la version précédente comme le service de délégation et la gestion de certificats.

GT a été conçu pour résoudre le problème d'héterogenieté des ressources (archives de données, ordinateurs, réseaux, politiques de sécurité) qui forme un obstacle pour la collaboration entre les organisations virtuelles. Ses services, interfaces et protocoles permettent aux utilisateurs d'accéder aux ressources distantes comme si elles sont en local, avec un contrôle sur qui est autorisé de les utiliser et quand **[Tga09]**.

La figure 2.1 ci-dessous montre l'évolution de l'architecture de GT dans le temps.

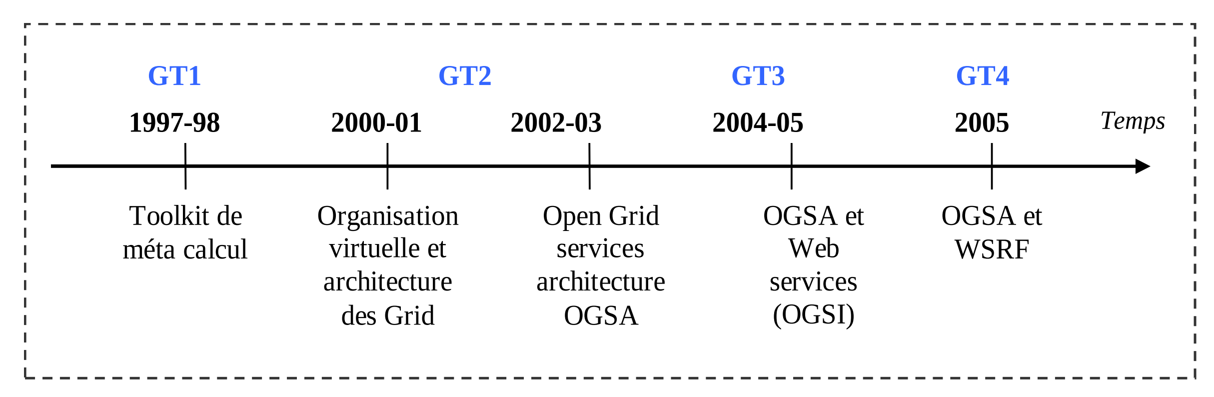


Figure 2. 1 Évolution de l'architecture de Globus

[3] Dans le reste de ce document nous utilisons le sigle GT pour désigner Globus Toolkit

2.2. ARCHITECTURE DE GLOBUS TOOLKIT 4 (GT4)

GT contient des services logiciels, des librairies pour la gestion et la découverte de ressources, la sécurité et la gestion des fichiers et des données. Il contient des composants qui peuvent être utilisés ensemble ou indépendamment les uns aux autres pour le développement d'applications.

GT4 fournit un ensemble de composants et des capacités, nous citons les implémentations de services, des outils pour la construction de nouveaux Web services en Java, C et Python, une infrastructure de sécurité basée sur des standards, des APIs pour les clients, et des programmes de lignes de commande pour l'accès aux différents services et capacités **[Fos06]**.

Chaque application a ses propres besoins, mais il existe des fonctions qui se répètent fréquemment **[Fos06]**, notamment la découverte de ressources, la configuration d'une ressource de calcul donnée pour l'exécution d'une application, le déplacement des données d'un site vers un autre, etc. Le rôle de Globus est de rendre disponible toutes ces fonctionnalités qui peuvent apparaître sous forme de composants ou services, comme nous allons le voir dans l'architecture de Globus.

Avant de présenter l'ensemble des composants de GT4 il sera utile de donner d'abord une vue globale de l'architecture de Globus, qui est une vue client/serveur tel que définit dans **[Fos05a]**. La figure 2.2 ci-dessous, présente l'architecture de GT4.

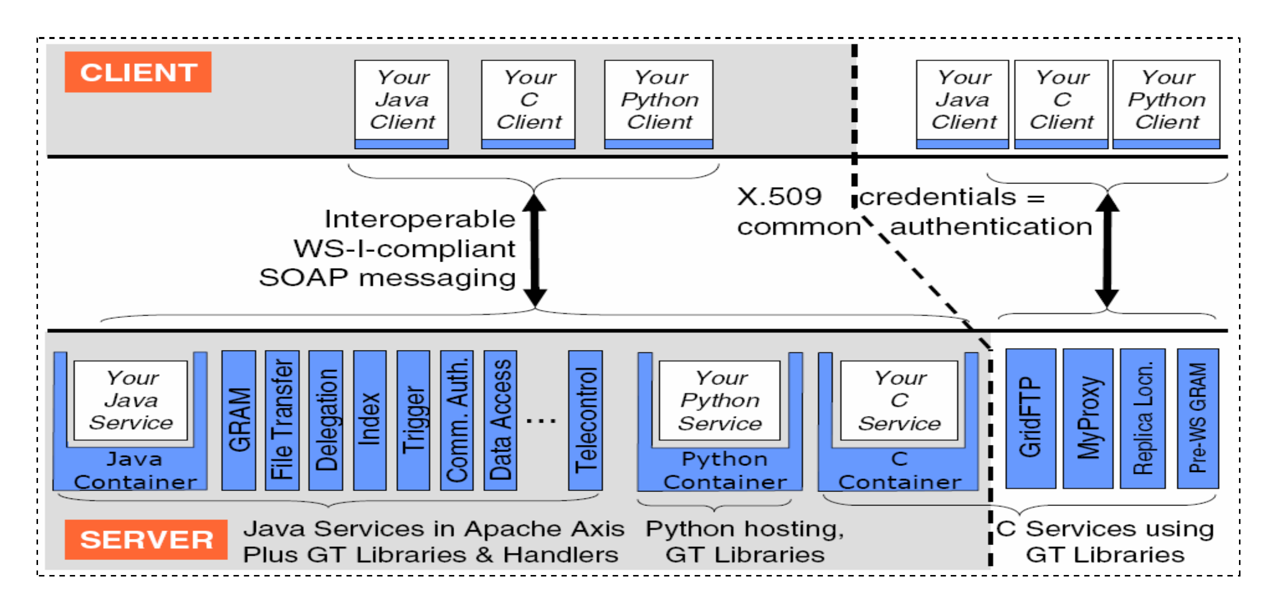


Figure 2. 2 Composants et interactions de GT4 (source[Fos05a]).

Les composants principaux de GT4 sont les trois suivants :

a) Dans la partie basse de cette architecture nous trouvons les implémentations de services d'infrastructure. Parmi ces services, nous citons la gestion de l'exécution (GRAM), déplacement et accès aux données (GridFTP, RFT, OGSA-DAI), gestion des répliques (RLS et DRS), gestion et découverte (Index, Trigger, WebMDS), gestion des autorisations et de certificats (MyProxy, Delegation, SimpleCA) et la gestion des instruments (GTCP) dont la plupart sont des services Java.

b) Trois conteneurs qui servent à héberger les services développés par les utilisateurs écrits respectivement en Java, C et Python. Ces conteneurs contiennent les implémentations des services qui sont nécessaires pour le développement des services utilisateurs tels que la sécurité, la gestion d'état et la découverte de la structure de Grid. Ils différent des environnements d'exécution par leur utilisation des spécifications des Web services notamment, WSRF, WS-Notification et WS-Security.

c) Un ensemble de librairies client permettant aux programmes client écrits en Java, C et Python d'invoquer les opérations des services du Toolkit et ceux développés par les utilisateurs.

Les clients peuvent interagir de la même façon avec tous les services grâce aux mécanismes et abstractions uniformes fournis par le Toolkit, ce qui permet la construction de systèmes complexes et interopérables et encourage la réutilisation de codes. Cette uniformité est du par exemple à l'utilisation d'une infrastructure de sécurité et de messagerie commune qui permet l'interopérabilité entre services et applications.

2.3. SERVICES ET MODULES DE GLOBUS

GT4 offre et implémente un ensemble de services qui peuvent être basés ou non sur les Web services, on parle alors des composants WS (WS-components) et des composants non WS (non WS-components). Selon la figure 2.3 montrée ci-dessous, quatre grandes classes de services peuvent être distinguées : la sécurité, la gestion de données, la gestion des jobs et le service d'information. La dernière colonne montrée dans la figure 2.3 s'agit des librairies et outils pour héberger les services existants et développer de nouveaux services comme expliqués dans la section précédente. Dans cette partie nous nous intéressons aux quatre services cités auparavant en essayant de montrer le rôle de chacun d'eux, les standards et technologies utilisés ainsi que leurs interactions.

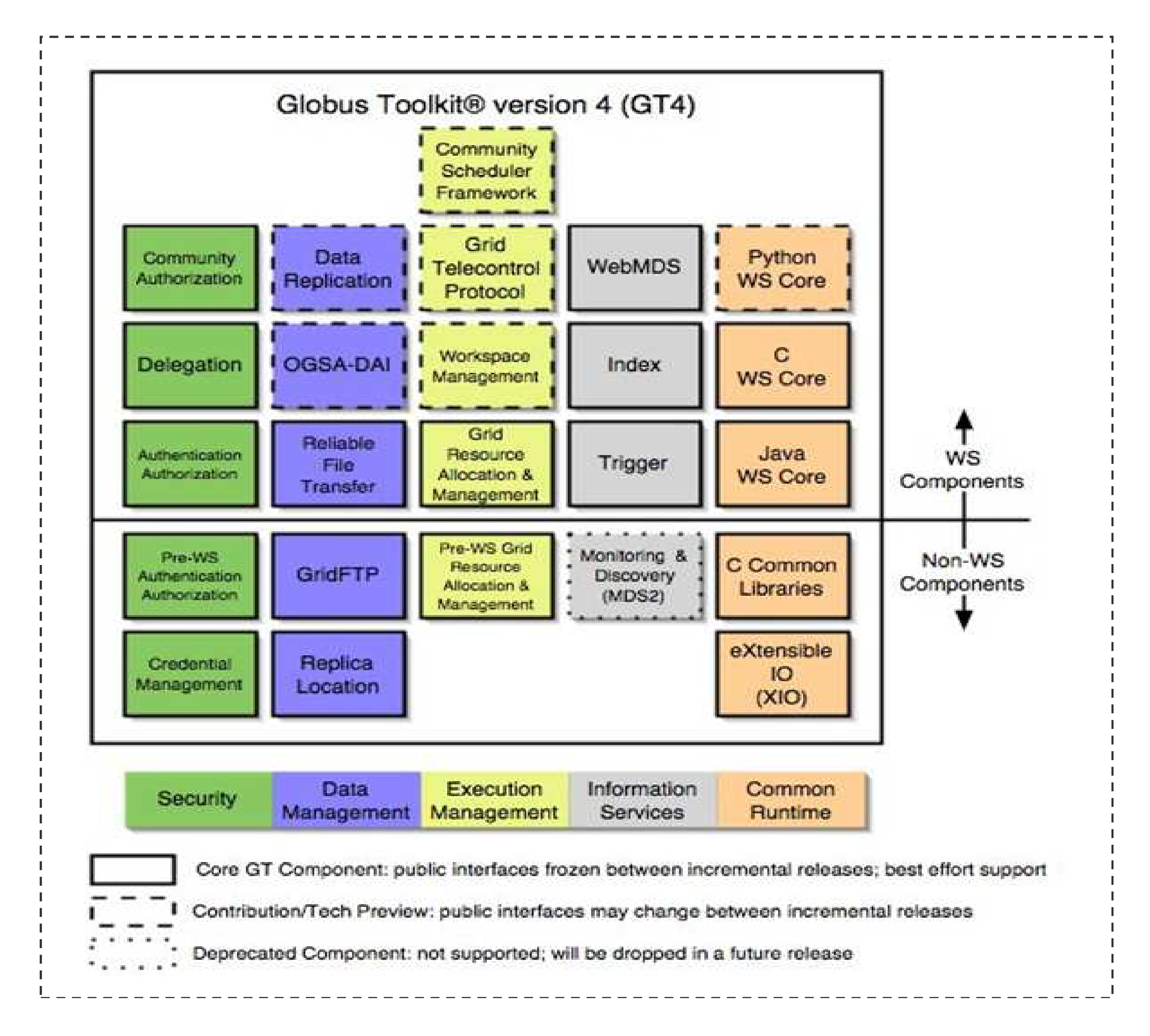


Figure 2. 3 Globus Toolkit 4 (source [Tga09])

2.3.1. LA GESTION DES DONNEES

Ce module permet le transfert des fichiers entre les nœuds de Grid ainsi que la gestion de ce transfert, pour cela Globus offre une collection de services dont chacun est responsable d'une tâche donnée ; la gestion de données distribuées, leur localisation et leur transfert. On distingue deux catégories de composants de gestion de données : le déplacement de données et la réplication de données.

2.3.1.1. Le déplacement de données

Deux composants de GT4 sont responsables du transfert de données notamment le GridFTP et le RFT.

GridFTP

GridFTP est un service de transfert de données entre les machines de Grid, il ne fournit pas une interface de Web service (il n'emploie pas SOAP, WSDL, etc.), c'est une extension du

protocole de transfert des fichiers FTP. Le protocole GridFTP permet un transfert sécurisé, robuste, rapide et efficace de données **[Tga09]**.

Le protocole GridFTP est adopté pour le transfert de données dans le Grid à cause de sa richesse et ses caractéristiques qui peuvent être résumés ainsi **[ABK+05]** :

- Contrôle Third-party de transfert de données: L'opération Third-party permet à un utilisateur ou à une application sur un seul site d'initier, contrôler et gérer le transfert de données entre deux autres sites. Il est nécessaire de disposer d'un contrôle authentifié third-party pour contrôler et gérer les transferts de données entre différents serveurs de stockage des communautés distribuées.
- Authentification, intégrité et confidentialité des données : GridFTP supporte l'authentification de GSS-API (interface fournissant des services de sécurité à utiliser dans des applications distribuées) et supporte aussi les niveaux de contrôle utilisateur de l'intégrité et/ou la confidentialité des données.
- Transfert de fichiers détachés : GridFTP définit des extensions pour le protocole qui permettent le transfert de données partitionnées entre plusieurs serveurs. Par conséquent, les fichiers peuvent être téléchargés en pièces simultanément à partir de sources multiples, ou sous forme de flots séparés et parallèles de la même source, ce qui augmente la vitesse de transfert.
- Transfert parallèle de données : L'utilisation de plusieurs flux TCP en parallèle entre une seule source et destination augmente et agrége la bande passante, le GridFTP supporte ce parallélisme via les extensions de commande FTP et les extensions du canal de données.
- Transfert partiel de fichiers : FTP permet de reprendre le transfert interrompu d'un fichier à partir d'un point précis du fichier. Le GridFTP permet et accepte des requêtes de transfert des parties d'un fichier. Cette caractéristique est utile pour des applications où seulement de petites sections d'un large fichier de données sont requises pour le traitement.
- La négociation automatique des tailles des tampons/fenêtres TCP. L'utilisation des paramètres optimaux des tailles des tampons/fenêtres influe grandement sur la performance de transfert de données, et le réglage automatique est une tâche difficile pour des utilisateurs non experts et peut provoquer de graves erreurs. GridFTP permet la négociation automatique et le réglage manuel des tailles du tampon TCP pour de gros fichiers ou de larges groupes de petits fichiers.

- Transfert fiable et rétablissement de l'état de transfert : FTP contient les caractéristiques de base pour reprendre les transferts échoués, le GridFTP exploite ces caractéristiques pour former son nouveau protocole qui permet de résoudre le problème des serveurs et la non disponibilité des réseaux. Un transfert peut être à tout moment automatiquement repris lorsque un échec survient.

GT4 fournit un serveur GridFTP et un client GridFTP qui sont implémentés respectivement par le démon in.ftpd et la commande Globus-url-copy, et supportent deux types de transfert :

- Le transfert standard de fichiers, où le client transfère un fichier local à une machine distante.
- Le transfert de fichiers Third-party, lorsque le client transfère un fichier se trouvant dans un espace de stockage distant vers un autre serveur distant.

Le service RFT (Reliable File Transfer)

RFT est un Web service basé sur WSRF qui utilise GridFTP pour transférer de grandes quantités de données. La motivation principale de ce service **[AFM04]** est le fait que :

- GridFTP ne fournit pas une interface de Web service,
- Un client voulant effectuer des transferts de données doit garder le canal de control ouvert pendant toute l'opération de transfert, ce qui n'est pas pratique quand il s'agit de longs transferts,
- GridFTP permet de rétablir ou reprendre le transfert lorsque un problème survient (panne réseau ou échec de serveur), mais lorsque le problème est au niveau de la machine du client il est impossible de rétablir le transfert car l'état du transfert se situe dans la mémoire du client.

Pour résoudre les problèmes cités dessus il est indispensable de disposer d'une interface d'un service basé sur les Web services et manipulant l'état (un Web service basé sur WSRF) qui permet de sauvegarder l'information d'état dans une entité de stockage fiable différente de celle du client, et c'est à partir de cette information que se fait le rétablissement du transfert. Ce service créé appelé RFT.

Pour soumettre une requête de transfert à RFT il suffit de lui indiquer la liste des URLs source et destination (gsiftp:// URLs), avec une possibilité de spécifier des paramètres tels que les flux GridFTP et la taille des tampons TCP. RFT stocke les requêtes de transfert des clients dans une base de données. Un client peut obtenir des informations concernant l'état du transfert, il peut

s'inscrire pour être notifié des événements du changement d'état. La figure 2.4 montre la relation entre RFT et GridFTP.

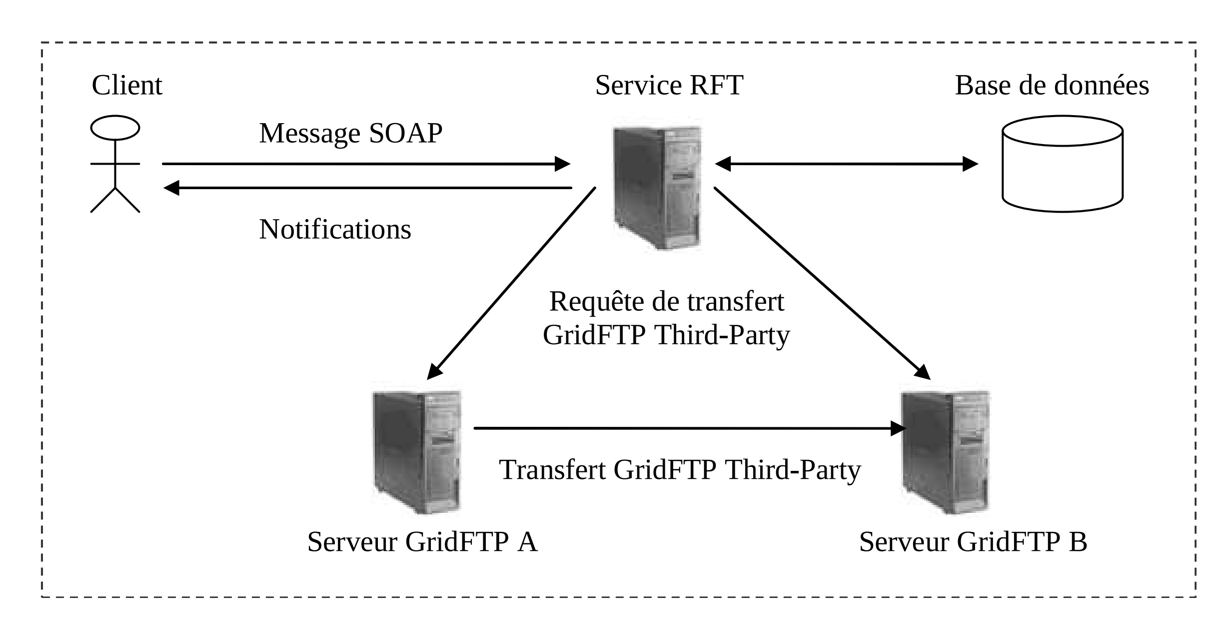


Figure 2. 4 La relation entre RFT et GridFTP (source [JBFT05])

2.3.1.2. La réplication de données

Replica Location Service (RLS) est un système décentralisé qui maintient et donne accès aux informations sur l'emplacement physique des fichiers et données répliqués **[CDF+02]**. Il est développé par les deux projets Globus et DataGrid **[GKL+02]**, il fournit des mécanismes pour enregistrer et localiser les répliques, il utilise pour cela des noms logiques et des noms de destination ou physiques.

- Un nom logique, identifie le contenu de données qui peut avoir une ou plusieurs répliques physiques
- Un nom de destination, c'est l'emplacement physique de la réplique ou bien des autres noms logiques représentant les données.

Le RLS contient deux composants principaux **[Tga09]** qui sont: LRC (Local Replica Catalogs) et RLI (Replica Location Indexes).

- Local Replica Catalog (LRC): contient les associations des noms logiques et les noms physiques des répliques. Le LRC est donc un catalogue qui permet à un client de découvrir les répliques associées avec le nom logique.
- Replica Location Indexe (RLI): agrége l'information d'état sur un ou plusieurs LRCs, cet outil est utile pour les utilisateurs et les applications qui ont besoin de trouver l'emplacement des fichiers dans un environnement de Grid. Un déploiement simple de

RLS consiste en un seul LRC qui se comporte comme un registre d'associations pour un ou plusieurs systèmes de stockage. Pour un déploiement distribué un deuxième composant est nécessaire qui s'agit d'un serveur appelé RLI, chaque serveur RLI collecte des informations sur les associations des noms logiques stockées dans un ou plusieurs LRCs (Chaque LRC envoie périodiquement des informations sur les associations de ses noms logiques à un ensemble de RLIs). Le RLI aussi répond aussi aux requêtes concernant ces associations. Lorsque un client veut découvrir les répliques qui peuvent exister dans différents emplacements le client va envoyer cette requête au serveur RLI plutôt qu'au LRC individuel, ensuite le RLI répond à cette requête en retournant la liste de tous les LRCs qui ont les associations pour les noms logiques contenus dans la requête. Le client alors demande aux LRCs de lui trouver les emplacements physiques des répliques. La figure 2.5 ci dessous montre un exemple de déploiement de RLS.

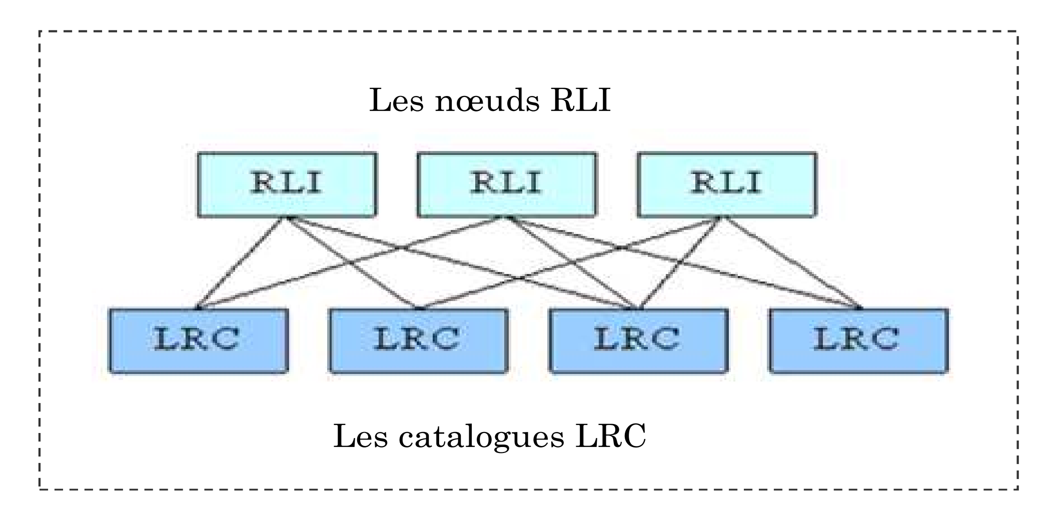


Figure 2. 5 Exemple sur le déploiement de RLS (source [CPB+04])

2.3.1.3. Les services de haut niveau

GT4 fournit un service de gestion de données de haut niveau qui est le DRS (Data Replication Service) qui utilise les deux services RLS et RFT **[Tga09]**.

Le service de réplication de données (DRS) **[CSK+05]** comme son nom l'indique, permet de répliquer un ensemble de fichiers dans un système de stockage local en assurant la découverte des répliques, ceci se fait par l'enregistrement des nouveaux fichiers dans des registres appropriés. Lorsque le service de réplication reçoit une requête il communique avec le RLS pour savoir où se situent les fichiers spécifiés dans le Grid et si elles existent dans le site local, ensuite après la découverte de leurs emplacement li communique et utilise le RFT pour effectuer l'opération de transfert des fichiers spécifiés par la requête vers le système de stockage local en se basant sur le

protocole GridFTP. Lorsque le transfert est achevé, le DRS enregistre la nouvelle réplique auprès du RLS pour que les nouvelles répliques créées puissent être découvertes à nouveau par d'autres sites.

DRS est implémenté comme un Web service basé sur WSRF, la ressource à état de ce service est appelée 'replicator' **[CSK+05]**, elle présente la requête de réplication envoyée par le client qui lui permet de connaître l'état de la requête et de contrôler son comportement. Parmi les propriétés de cette ressource nous citons : l'état de la requête (actif, en attente, suspendu, terminé ou détruit), le stade ou étape courante de la réplication (découverte, transfert ou enregistrement), résultat final de la réplication (aucun, fini, échec ou exception), 4) et finalement le nombre total des fichiers (terminés, finis, ou échoués) dans la réplication.

Lorsque un client envoie une requête de réplication au DRS ceci permet la création du 'replicator' ou bien la ressource à état et à ce moment là, le client peut contrôler l'opération de réplication tel que, lancer, suspendre et arrêter l'opération de réplication. Finalement après terminaison de la réplication la ressource à état doit être détruite ainsi que ses propriétés.

DRS utilise pour son fonctionnement en plus de RFT et RLS deux autres services, le service de délégation qui permet l'authentification et offre des APIs d'autorisation pour communiquer de manière sécurisée avec d'autres services et authentifier les utilisateurs, et le service GridFTP qui permet le transfert des données entre les sites. Le rôle de DRS est alors l'assurance de l'existence d'un fichier donné dans un site de stockage.

2.3.2. LA GESTION DES TACHES

Un utilisateur voulant exécuter une tâche sur le Grid aura besoin d'outils qui lui permettent de spécifier les capacités nécessaires pour l'exécution de sa tâche (vitesse processeur, taille mémoire, ..), allouer les ressources requises pour l'exécution, lancer l'exécution et vérifier l'évolution et l'état des jobs, et enfin récupérer les résultats. Toutes ces fonctionnalités sont offertes par le gestionnaire des tâches GRAM (Grid Resource Allocation and Management) en fournissant une interface de Web service pour l'initiation et la gestion de l'exécution sur des nœuds distants.

GRAM n'est pas un ordonnanceur de ressources mais plutôt un protocole permettant la communication avec différents ordonnanceurs de ressources locaux en utilisant un format de message standard **[Tga09]**. L'implémentation de GRAM offerte par GT4 inclut des interfaces pour des ordonnanceurs tel que PBS, CONDOR, LSF et SGE et aussi un ordonnanceur 'fork' qui exécute les jobs immédiatement et les contrôle.

Le Grid basé sur le middleware Globus contient plusieurs GRAMs, chacun est responsable d'un ensemble particulier de ressources **[FK98]**.

GT4 inclut deux services GRAM différents : 'pre-WS GRAM' appelé aussi GRAM2 introduit dans GT2 et 'WS GRAM' appelé GRAM4 basé sur les Web services et introduit dans GT4. GRAM4 fournit de meilleures fonctionnalités par rapport à GRAM2, il est beaucoup plus performant lorsqu'il s'agit des soumissions concurrentes des jobs, mais il est moins rapide que GRAM2 pour les soumissions séquentielles **[FFM07]**. GRAM utilise différents types de services pour la gestion des jobs :

- Services de gestion des jobs, qui servent pour le contrôle et la gestion du cycle de vie des jobs.
- Services de transfert des fichiers, qui permettent le transfert des fichiers entre les ressources de calcul.
- Services de gestion des certificats, utilisés pour contrôler la délégation des droits entre les différents éléments de GRAM selon les besoins des applications.

GRAM4 ne fonctionne pas seul pour permettre l'exécution distante des jobs et leur gestion, il utilise pour cela un certain nombre de composants qui sont disponibles dans GT4 **[Exe09]** :

- Le service RFT est invoqué pour effectuer le transfert des fichiers avant ou après exécution des jobs (file staging), le RFT permet de rétablir le transfert des fichiers en cas d'échec (dans GRAM2 si l'opération de staging est échouée l'exécution du job est annulée).
- GridFTP permet le transfert des fichiers entre les ressources de calcul et les éléments de stockage de données externes, il permet aussi de gérer le contenu des fichiers écrits par le job durant son exécution.
- Le service de délégation peut être utilisé lors de la soumission d'un job au service GRAM pour par exemple, accéder aux données se trouvant dans un élément de stockage distant où l'authentification est requise. Dans GRAM2 la délégation des droits est exprimée à travers la requête, dans GRAM4 la délégation se fait à travers une interface **[FFM07]**.

Deux autres types de composants sont utilisés par GRAM4 : composants internes et composants externes.

- Les composants internes se résument en deux composants : le générateur d'événement d'ordonnanceur "Scheduler Event Generator" qui donne au service GRAM4 la

possibilité de contrôler les jobs et le ‘Fork Starter’ qui lance et contrôle l’exécution des jobs.

- Les composants externes sont deux : l’ordonnanceur de job local (Local job scheduler) qui permet de gérer les ressources d’un élément de calcul et “Sudo“ qui permet de faire fonctionner le GRAM sous un compte utilisateur particulier sans utiliser tous les privilèges de l’utilisateur.

GRAM4 utilise le langage de description des jobs *Job Description Document (JDD)* qui est une extension du *Resource Specification language (RSL)* à base de l’XML, avec ce dernier on peut spécifier le fichier exécutable et les ressources vérifiant les conditions.

2.3.3. LE SERVICE D’INFORMATION

Le Grid est caractérisée par sa nature dynamique, sa structure et son état changent avec le temps, il est donc indispensable de disposer d’outils ou de composants qui nous donnent des informations sur les changements (structure et état) subis par le système. Les informations collectées sont utilisées soit pour découvrir de nouveaux services ou ressources ou pour permettre la gestion et le contrôle de l’état du système. Les mécanismes de découverte et de gestion sont offerts par Globus et implémentés par le composant *Monitoring and Discovery Services (MDS)*. Dans Globus, chaque service doit fournir des informations qui semblent être utiles pour l’utilisateur **[FK98]**, par exemple, les deux services GRAM et RFT fournissent des informations qui sont collectées par les services de MDS.

Comme pour le gestionnaire des tâches GRAM, GT4 est composé de deux versions de MDS : MDS2 (Pre-WS component) c’est l’ancienne version, il n’est pas basé sur les Web services et MDS4 qui est un composant WS (WS component). Dans cette section nous allons présenter le MDS4 mais il faut d’abord montrer les différences clés entre ces deux versions.

Comme expliqué dans **[Fos05b]**, le MDS4 a les mêmes caractéristiques que les versions précédentes mais il ne peut pas interopérer avec eux. Les différences importantes sont :

- Langage de requête plus puissant (Xpath au lieu de Lightweight *Directory Access Protocol (LDAP)*).
- Nombre de composants plus réduit ce qui rend l’implémentation plus simple et plus robuste.
- Configuration plus simple à cause du nombre réduit des composants et sa faible intégration avec l’implémentation de GT4.
- Une interface appropriée aux différentes sources d’information à cause de l’extensibilité de son architecture.

- Pas besoin pour de schémas prédéfinis pour les fournisseurs d'information.
- Une performance inférieure à celle de MDS2 car les technologies utilisées ne sont pas encore matures, et dans des cas il est plus performant à cause des changements architrecturals, tel que la suppression des requêtes en cascade.

La découverte et le contrôle sont deux opérations qui nécessitent de collecter des informations à partir des sources d'informations distribuées. Par conséquent, le MDS utilise des services d'agrégation (aggregator services) qui collectent les informations d'état récentes à partir des sources d'informations et des interfaces de navigateurs, des lignes de commande et des interfaces de Web service qui permettent aux utilisateurs d'accéder aux informations collectées.

2.3.3.1. Les services d'agrégation (Aggregator services)

Le Service d'Index

C'est le service fondamental de MDS et il existe par défaut dans chaque conteneur de GT4, il sert à collecter les informations provenant de différentes sources ensuite il permet leur publication. Le langage de requête utilisé pour accéder aux informations qu'il expose est le Xpath.

Le service d'index est un registre similaire à UDDI, mais plus flexible **[Tga09]**. Les données collectées sont disponibles sous forme de documents XML où les données sont maintenues comme des propriétés de ressources WSRF **[Gt309]**. Un client voulant accéder aux informations accède au service d'Index à travers les requêtes standards des propriétés et les interfaces de notification et d'abonnement. Chaque service d'index peut s'enregistrer dans un autre service d'index et ainsi de suite de manière hiérarchique dans le but d'agréger les données dans différents niveaux. Chaque entrée de ce service a une durée de vie et sera supprimée si elle n'est pas rafraîchie après qu'elle expire **[Tga09]**.

Le Service Trigger

Collecte des informations sur les ressources de Grid et il peut être configuré pour l'exécution d'un programme si les données collectées répondent à certaines conditions. Le trigger compare les données collectées avec des conditions définies dans un fichier de configuration, lorsque la condition est vérifiée, une action est exécutée, par exemple envoyer un mail à un administrateur système lorsque l'espace disque sur un serveur donné atteint un seuil **[Tga09]**.

2.3.3.2. Aggregator Framework

C'est le framework sur lequel sont construits les deux services d'agrégations (index et trigger), c'est une infrastructure qui fournit des interfaces et des mécanismes communs pour

interagir avec les sources de données **[Mds09]**. L'aggregator collecte les données à partir d'une source d'agrégation, il peut être interrogé par plusieurs méthodes, par des requêtes lancées ou des notifications demandées par des WS-Services ou des programmes pour exécuter des actions lancées après vérification de certaines conditions.

2.3.3.3. Les Fournisseurs d'Information

Les fournisseurs d'information sont utilisés pour collecter les informations à partir de sources spécifiques. Ces fournisseurs forment des interfaces pour d'autres systèmes et outils, comme le contrôleur de cluster Ganglia, les ordonnanceurs PBS et Condor, etc. Chaque fournisseur d'information offre des informations différentes, nous citons ici quelques types de fournisseurs :

- Le fournisseur d'information *Ganglia*, contient les données sur les clusters et les publie sous formes de propriétés de ressources, parmi les informations publiées nous avons, le nom et l'identificateur d'un hôte, taille mémoire, version et nom du système d'exploitation, et d'autres données.
- GRAM, publie des informations sur un ordonnanceur local (nombre de CPUs disponibles et libres, informations sur la fille d'attente, le nombre de jobs, etc.)
- RFT, publie des informations sur l'état du serveur, l'état du transfert pour un ou plusieurs fichiers, nombre de transferts actifs et quelques informations sur la ressource exécutant le service.
- *Community Authorization Service (CAS)*, publie des informations qui identifient l'organisation virtuelle qu'il dessert.

2.3.3.4. L'interface utilisateur WebMDS

Le client WebMDS présente les informations du service d'index dans un navigateur Web standard **[Tga09]**. Il utilise les transformations XSLT pour décrire comment les propriétés de ressources du service d'index sont converties en HTML **[Fos05b]**, les résultats sont affichés à l'utilisateur sous plusieurs formats. WebMDS utilise les requêtes standards des propriétés des ressources pour acquérir les données.

2.3.4. L'INFRASTRUCTURE DE SECURITE (GSI)

C'est l'infrastructure sur laquelle sont basés tous les autres services et composants de Globus. Au départ cette infrastructure a été développée pour résoudre un problème unique **[FK98]**, celui de l'authentification qui est le mécanisme par lequel une entité vérifie l'identité

d'une autre, et à partir duquel d'autres services de sécurité peuvent être construits comme l'autorisation et l'encryption.

Le GSI utilise la cryptographie à clé publique (connue aussi sous la cryptographie asymétrique) comme une base pour son fonctionnement. Le certificat est le concept clé de l'authentification offerte par le GSI, chaque utilisateur et chaque service est identifié via le certificat. Le certificat contient plusieurs champs d'informations à savoir ; le nom de sujet (identifie l'objet que le certificat représente), la clé publique appartenant au sujet, l'identité de l'autorité de certification *Certification Authority (CA)* qui a signé le certificat et qui certifie que le couple clé publique et identité appartient au sujet, et la signature numérique de la CA. Les certificats du GSI sont encodés au format X.509.

Comme montré dans la figure 2.5 ci-dessous, le GSI fournit quatre fonctions principales **[Wel05]** : protection de messages, authentification, délégation et autorisation.

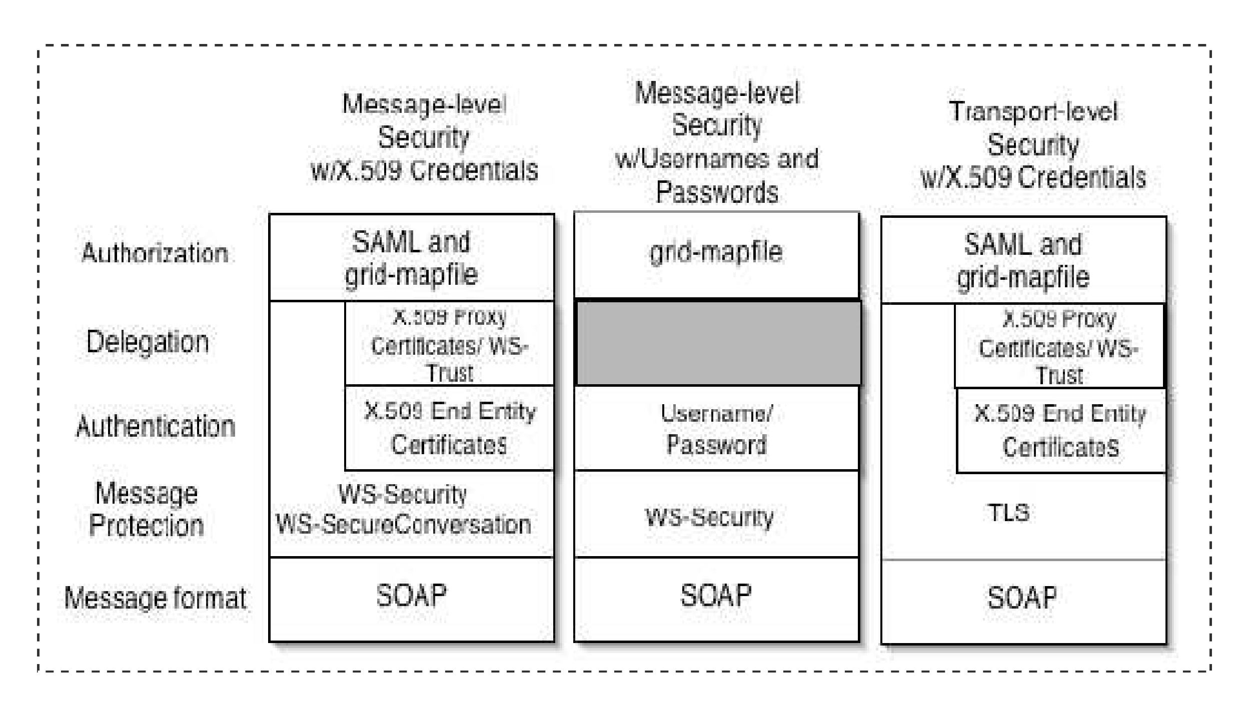


Figure 2. 6 Une vue d'ensemble de l'infrastructure GSI (source [Wel05])

2.3.4.1. Protection de messages

GT4 fournit deux niveaux de sécurité : la sécurité au niveau message à travers l'utilisation du standard WS-Security et la spécification WS-SecureConversation pour fournir la protection des messages SOAP, et la sécurité au niveau transport via la couche de transport *Transport Layer Security (TLS)* qui permet l'authentification en utilisant les certificats Proxy X.509.

- *La sécurité au niveau transport* : utilisée en conjonction avec les certificats X.509 pour l'authentification, comme elle peut aussi être utilisée sans ces derniers pour la protection des messages. Le GSI utilisé soit avec le nom d'utilisateur/mot de passe ou avec les certificats X.509, utilise le WS-Security pour l'authentification.

- *La sécurité au niveau message* : le standard WS-Security et la spécification WS-SecureConversation sont implémentés par le GSI pour permettre la protection des messages SOAP. Le WS-Security permet d'assurer une sécurité des messages SOAP individuels (indépendamment des émetteurs et récepteurs). La Conversation sécurisée est uniquement offerte par le GSI lorsque ce dernier utilise les certificats X.509 pour l'authentification. Lorsque le GSI est utilisé avec les certificats X.509 il utilise WS-Security et WS-SecureConversation pour fournir des mécanismes de protection supplémentaires : Protection de l'intégrité, cryptage et prévention répétée.

2.3.4.2. Authentification et délégation

La délégation et l'authentification se font en général à travers les certificats X.509 et les clés publiques. Dans Globus le GSI introduit une nouvelle option c'est l'authentification à l'aide du nom d'utilisateur et le mot de passe **[Wel05]**.

GSI utilise les certificats X.509 pour identifier les entités (utilisateurs, services…). Le GSI supporte la délégation et le *Single Sign-On (SSO)* à travers l'utilisation des certificats proxy. Le proxy permet aux utilisateurs de déléguer leurs droits pour d'autres entités, c'est un nouveau certificat avec une nouvelle clé privée. Le proxy est signé par son propriétaire plutôt qu'avec le CA. En effet, Pour pouvoir accéder à plusieurs ressources ou applications par une seule authentification, l'utilisateur crée un proxy qui va agir en son nom.

2.3.4.3. Autorisation

L'autorisation est un mécanisme très important dans le GSI, il indique la personne autorisée à effectuer certaines tâches. C'est l'étape qui vient juste après l'authentification ; une fois un utilisateur est authentifié sur le Grid il est nécessaire de déterminer à quoi l'utilisateur est autorisé de faire (ressources à utiliser, taux d'utilisation, etc.)

La façon usuelle pour l'autorisation est l'utilisation d'une liste de contrôle d'accès qui contient les identités des personnes autorisés et le type d'accès permis. Le Grid utilise une approche similaire qui est le « grid-map file », ce fichier contient la liste des utilisateurs autorisés pour l'accès. Le GSI utilise un autre outil en plus du premier, c'est le standard *Security Assertions Markup Language (SAML)*.

2.4. CONCLUSION

Dans ce chapitre, nous avons présenté l'intergiciel Globus dans sa version 4 (GT4), son architecture et ses principaux composants ; gestion de données, gestion de l'exécution, service d'information et finalement, l'infrastructure de sécurité. Il existe d'autres services de haut niveau qui font partie de l'architecture de Globus mais qui n'ont pas été présentés parce que la plupart de ces services ne sont pas encore matures et ils sont en cours de développement, nous citons par exemple : OGSA-DAI -dans le module de gestion de données- fournissant une solution pour l'accès et la gestion des données distribuées, CAS (Community Authorization Service) dans le module de sécurité, Grid Telecontrol Protocol (GTCP) qui est utilisé pour le contrôle d'instrumentation, etc.

Notre contribution dans ce chapitre est le résumé de ce grand et complexe intergiciel dans quelques pages et à partir d'un nombre très important de références. Nous avons choisi un niveau de granularité moyen dans la description des différents composants de Globus pour rendre facile à assimiler ses divers concepts.

En conclusion :

- Un très grand nombre de technologies et de standards sont utilisés dans l'intergiciel Globus pour répondre aux besoins de Grid et plus précisément aux besoins des organisations virtuelles.
- Globus contient des services logiciels et des librairies pour la gestion des ressources, la sécurité, la découverte et la gestion des fichiers, qui sont nécessaires pour les applications à base de Grid.
- Globus est un logiciel en open source, et chaque composant peut être utilisé indépendamment de l'autre selon les besoins des utilisateurs, ce qui augmente le taux de son utilisation et de l'intégration avec d'autres solutions pour résoudre de nouveaux problèmes.
- GT4 offre et implémente un ensemble de services qui peuvent être basés ou non sur les Web services, et il contient aussi l'implémentation de WSRF.
- L'évolution des technologies a une grande influence sur l'évolution de l'architecture de Globus et plus particulièrement la convergence des technologies de Grid et des Web services, et l'apparition de WSRF.

- L'inconvénient majeur de cet intergiciel est dans son processus d'installation qui est une tache assez complexe, et malgré ça nous l'avons installé avec succès, par conséquent, un guide d'installation a été inséré à la fin de ce mémoire.



CHAPITRE 3

E-LEARNING ET COLLABORATION

« L'Éducation est un processus social ; l'éducation est un développement ; L'éducation n'est pas une préparation à la vie mais la vie elle-même »
John Dewey

Ces dernières années, avec l'évolution rapide des technologies d'information et de communication, une nouvelle forme d'apprentissage en ligne dite E-learning est apparue et devient actuellement un des moyens pédagogiques prometteurs. Afin de réaliser des systèmes de E-learning, plusieurs ressources sont requises tels que les contenus d'apprentissage, les ressources matérielles et humaines. Cependant, une communauté donnée ne peut assurer ce nombre important de ressources lorsqu'elle travaille de manière individuelle, par conséquent la collaboration avec d'autres communautés qui partagent des objectifs similaires s'avère importante.

Ce chapitre fait un tour d'horizon rapide sur plusieurs concepts liés au E-learning, apprentissage et travail collaboratif, les communautés virtuelles d'apprentissage et les environnements de collaboration.

En effet, l'objet de ce chapitre n'est pas de présenter uniquement les concepts fondamentaux du E-learning et de collaboration, mais il met en exergue la relation existante entre le Grid - présenté dans le premier chapitre - et les concepts de E-learning, ceci est réalisé à travers une synthèse de quelques travaux. Nous considérons alors qu'une bonne partie de ce chapitre est une justification de notre problématique et proposition.

3.1. INTRODUCTION

L'apparition de l'informatique a permis l'avènement d'une nouvelle forme d'enseignement dite 'E-learning', qui est devenu actuellement un des moyens pédagogiques prometteurs. L'objectif principal du E-learning est d'améliorer la qualité de l'apprentissage et non de se substituer aux modes traditionnels d'enseignement **[MAD05]**. Aujourd'hui avec l'évolution des technologies d'information et de communication une nouvelle forme de travail collectif dite *«communautés virtuelles d'apprentissage»* est apparue. Ce sont des communautés réelles qui comprennent de vraies personnes, des enjeux importants et de nombreuses ressources. Un caractère commun de ces communautés est la collaboration et le travail en commun via le partage de ressources de différentes natures (matérielles logicielles et humaines).

Les concepts du E-learning, apprentissage collaboratif et communautés virtuelles d'apprentissage seront abordés dans ce présent chapitre. Afin de mieux montrer la liaison existante entre ce chapitre et les chapitres précédents, ainsi que sa relation avec la partie contribution, nous allons présenter un ensemble de travaux qui montrent cette relation en précisant d'abord quel sera l'apport de Grid pour le E-learning et pour la collaboration.

3.2. LE E-LEARNING

Dans cette partie seront abordés de manière très concise les concepts fondamentaux du E-learning notamment ses composants, acteurs, les activités qu'il offre, et un aperçu sur les plateformes existantes.

3.2.1. QU'EST CE QUE LE E-LEARNING ?

Plusieurs mots sont utilisés pour traduire le terme E-learning notamment, apprentissage en ligne, formation en ligne, enseignement à distance, etc., mais le plus adopté est l'apprentissage en ligne. En général, le terme 'E-learning' désigne l'utilisation d'Internet dans le cadre d'une formation.

Il existe une grande variété de définitions, cependant il n'y a pas une définition commune **[Dub03]**. Certaines sont centrées sur le support technologique, d'autres sur la distance. Certaines insistent sur les aspects pédagogiques, le type d'interaction ou le mode de tutorat, tandis que d'autres proposent une synthèse. Nous préférons celle mettant l'accent sur le média technologique. Nous discutons la suivante définition qui est donnée par la commission européenne :

Utilisation des nouvelles technologies multimédia et de l'Internet pour améliorer la qualité de l'apprentissage en facilitant l'accès à des ressources et des services, ainsi que les échanges et la collaboration à distance.

A partir de cette définition nous distinguons les points suivants :

- Le but final de tout système de E-learning est l'amélioration de la qualité d'apprentissage, pour réaliser ceci ces systèmes sont basés sur un ensemble de technologies comme l'Internet qui offre de nombreuses fonctionnalités nécessaires pour réaliser un apprentissage à distance tel que : le transfert de fichiers, la recherche d'informations, les services de communication et aussi les technologies multimédia.
- Quatre opérations sont énumérées et considérées comme indispensables pour tout système d'apprentissage en ligne notamment :
  - L'accès aux ressources (contenu d'apprentissage, par exemple),
  - L'accès aux services (le chat, la vidéoconférence, …),
  - Les échanges d'informations et de connaissances entre enseignants/apprenants apprenants/apprenants.
  - La collaboration à distance entre les différents membres qui font partie du système de E-learning à travers les outils de collaboration, et c'est ce point qui fait l'objet de notre recherche.

Plusieurs terminologies sont utilisées pour désigner le même concept mais avec des nuances, entre autres :

- Formation A Distance (FAD).
- Enseignement A Distance (EAD).
- Formation Ouverte et A Distance (FOAD).
- E – Formation.
- Formation Ouverte.

3.2.2. COMPOSANTS D'UN DISPOSITIF E-LEARNING

En général, quatre composants principaux peuvent être distingués dans les systèmes de E-learning **[PV03]** : les personnes, le système auteur, système d'exécution, et LMS (Learning Management System). Les personnes dans ces systèmes peuvent être des auteurs (enseignants ou concepteurs) ou des apprenants, et d'autres introduisent les formateurs et les administrateurs.

Les auteurs créent le contenu, ce dernier est stocké sous le contrôle d'un LMS et généralement dans une base de données. Le LMS lui même est géré sous le contrôle d'un administrateur et interagit avec un environnement d'exécution qui est adressé par les apprenants.

Le LMS permet aux administrateurs de gérer les différentes tâches liées aux processus de E-learning.

3.2.3. LES ACTEURS D'UN SYSTEME DE E-LEARNING

Plusieurs acteurs peuvent faire partie d'un système de E-learning, et plusieurs classifications existent. Dans ce qui suit nous allons les classifier, selon leur rôle, en trois catégories principales : les apprenants, les enseignants et les administrateurs qui sont illustrés dans la figure 3.1 présentée ci-dessous :

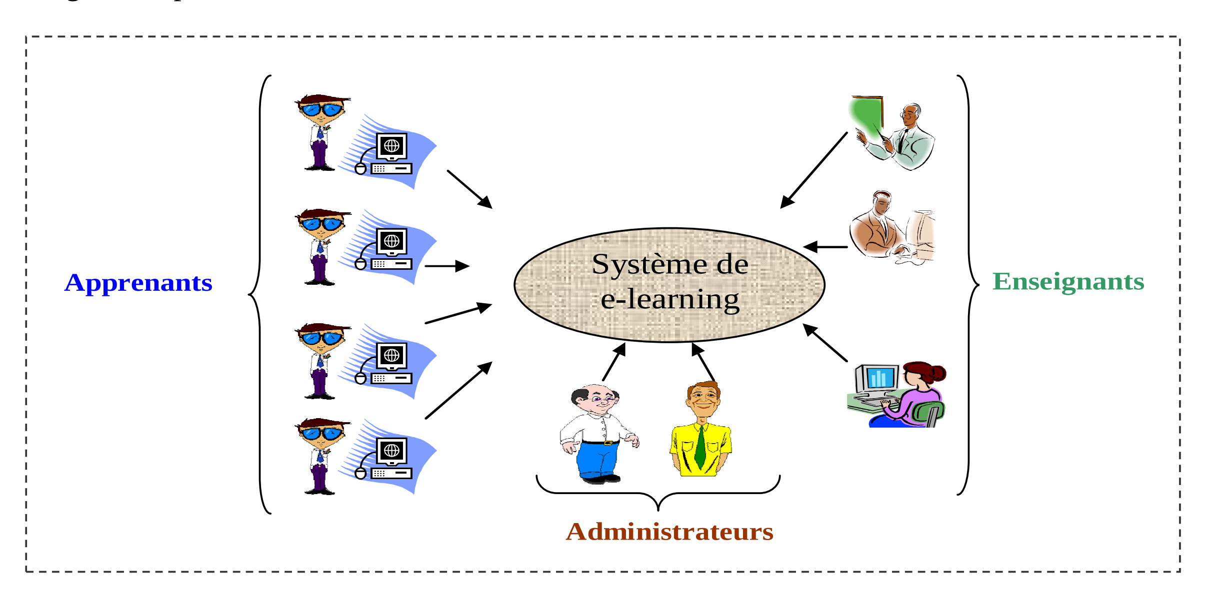


Figure 3. 1 Les acteurs d'un système de E-learning

3.2.3.1. Apprenant

C'est la personne concernée par l'apprentissage et voulant acquérir des connaissances en s'inscrivant à la plateforme. Chaque apprenant possède un ensemble d'informations qui le caractérisent, regroupées dans une entité appelée "profil". Plusieurs activités peuvent être exercées par l'apprenant, à savoir la participation aux forums, la consultation des cours, la communication avec les autres apprenants, etc.

3.2.3.2. Enseignant

C'est l'élément responsable de la diffusion de la connaissance, la création de cours et la proposition des tests et des quiz, etc. Un enseignant a plusieurs rôles ; pour cette raison, il existe plusieurs catégories d'enseignants, citons par exemple :

- L'auteur : responsable de la création des cours selon les objectifs pédagogiques définis en utilisant les outils disponibles sur la plateforme.
- Le tuteur : joue le rôle d'un superviseur pour l'activité d'apprentissage, il suit les apprenants, les anime, les évalue, etc.

- L'évaluateur : comme son nom l'indique, il permet l'évaluation et le suivi des apprenants en créant et gérant les tests d'évaluation.

3.2.3.3. Administrateur

Son rôle est lié principalement à la partie technique de la plateforme. Il gère aussi les comptes, affecte les droits d'accès aux acteurs… **[ACG05]**. Il s'occupe de tous les apprenants et tous les enseignants **[ACG05]**.

3.2.4. OBJETS D'APPRENTISSAGE

Un objet d'apprentissage, aussi appelé Learning Object (LO), est une entité numérique ou non, qui peut être utilisée, réutilisée ou référencée lors d'une formation, dispensée à partir d'un support technologique. Il peut s'agir d'une composante concrète de l'environnement (ressource de manipulation de connaissance, service, outil) ou d'un scénario décrivant à priori ou à posteriori le déroulement d'une situation d'apprentissage **[BTNK07]**.

3.2.5. APERÇU SUR LES PLATEFORMES DE E-LEARNING

Une plateforme de E-learning, appelée aussi plateforme d'apprentissage en ligne est un outil de diffusion et de gestion de connaissances, associant des contenus de cours à des moyens de communication, des outils d'entraînement et d'évaluation **[KMS04]**. Les plateformes sont très nombreuses et offrent un choix très varié de services pour les usagers, elles permettent entre autres de :

- Regrouper un ensemble de fonctionnalités sur une même plateforme afin de supporter les différents acteurs de la formation (apprenants, enseignants, tuteurs, cours, etc.).
- Organiser et faciliter la création collaborative de contenu.
- Gérer les apprenants et leurs cursus.
- Gérer de multiples contenus.
- Gérer simultanément plusieurs formations.

Les outils de E-learning selon V.Baudin dans **[BDVT04]**, peuvent être répartis en deux catégories: Learning Mangement System (LMS) et Content Management System (CMS). Les systèmes purement LMS soutiennent les cours en ligne avec des accès distants, alors que les environnements purement CMS sont utilisés pour la gestion et la création hors ligne de cours. Aujourd'hui, de nombreux systèmes de E-learning supportent les deux fonctions à la fois, ils sont regroupés dans Learning Content Management System (LCMS). Dans ce qui suit, nous allons présenter chacun de ces systèmes :

3.2.5.1. LMS (Learning Management System)

Appelé aussi système de gestion d'apprentissage, c'est un logiciel intégré **[Lec07]** permettant la gestion des espaces de cours et des utilisateurs, le contrôle des accès et fournit un nombre important de fonctionnalités nécessaires pour un apprentissage et un enseignement en ligne à savoir : les outils de communication synchrones et asynchrones, outils d'évaluation, outils de travail collaboratif, diffusion des contenus pédagogiques et suivi des apprenants.

3.2.5.2. CMS (Content Management System)

Appelé aussi "système de gestion du contenu", permet la création et la gestion du contenu en ligne. Un contenu est composé d'un ensemble d'informations appelées objets d'apprentissage dans le contexte du E-learning **[BTNK07]**. Le CMS soutient également de manière efficace l'apprentissage à distance en raison de son application robuste des forums de discussion **[Sha07]**.

3.2.5.3. LCMS (Learning Content Management System)

Appelé aussi "système de gestion de contenus de formation", crée, valide, publie et gère les contenus d'apprentissage en combinant les fonctionnalités du LMS et CMS. Le LCMS peut fournir des fonctionnalités aux développeurs et aux utilisateurs à la fois **[Lea09]**. Il permet aux développeurs de créer, stocker, gérer et délivrer le contenu d'apprentissage numérique à partir d'un répertoire d'objets central. Il permet aux utilisateurs de créer, gérer, rechercher et réutiliser les objets d'apprentissage.

Le tableau ci dessous résume les fonctionnalités des différents systèmes de gestion **[Lms09, Sha07]** :

Tableau 2. 1 Quelques fonctionnalités des systèmes de gestion

LMS	CMS	LCMS
- Gestion des apprenants - Suivi du cursus de formation - Suivi et bilan de la formation - Outils simples de conception des cours - …	- Affichage en ligne de matériel de cours - Évaluation de l'apprenant - Forum de discussion - Communication - …	- Création de contenu - Stockage de contenu - Diffusion dynamique - Gestion du contenu - …

3.2.6. ACTIVITES OFFERTES PAR UN SYSTEME DE E-LEARNING

De manière générale, chaque dispositif de E-learning et plus particulièrement chaque plateforme d'apprentissage propose un ensemble d'activités synchrones, asynchrones,

collaboratives ou de gestion des connaissances, ces services sont, la plupart du temps, utilisés à l'intérieur d'une communauté, entre les différentes entités humaines qui la composent. Le tableau ci-dessous fait un récapitulatif sur les activités qui peuvent être fournies :

Tableau 2. 2 Activités d'un dispositif E-learning

Activités synchrones	Activités asynchrones	Outils collaboratifs	Outils pédagogiques
- Service de chat - Messagerie instantanée - Téléconférence - Voix sur IP	- Forum - Messagerie - Espaces du dépôt de fichiers - Calendrier et planification des événements - Tests effectués par les apprenants - Wiki - FAQ	- Tableau blanc - Eportfolio - Blog - Bureau virtuel - Wiki - Calendrier - Agenda - Forum	- Quiz - Sondage - Blog - Outils de gestion de connaissances - RSS - Outils de conception de parcours pédagogiques - Outils de développement et d'intégration de cours en ligne.

A partir de ce tableau, nous concluons tout d'abord que les outils utilisés dans un système de E-learning sont si nombreux, et que chacun a son propre rôle. Les mêmes outils peuvent appartenir à plus d'une catégorie ; le forum, par exemple, est une activité synchrone et en même temps c'est un outil collaboratif.

Ensuite, si nous mettons l'accent sur le service "*Calendrier et planification des événements*" qui présente une *activité asynchrone*, nous constatons que le rôle du calendrier est la planification des événements qui peuvent avoir lieu pendant une période de temps bien déterminée dans le système. Mais, que disons-nous sur la possibilité de fournir des informations en temps réel sur toutes les activités et les changements subis par le système sans aucune planification au préalable? Et si c'est possible de le faire, nous aurons besoin certainement d'un espace de stockage très important pour sauvegarder ces informations et une capacité de traitement assez importante ; par conséquent quelle est la technologie qui peut apporter une solution à ce problème ? Cette question forme un point important dans notre proposition et sera abordée dans la partie contribution.

3.2.7. A QUI S'ADRESSE LE E-LEARNING ?

Le E-learning est souvent employé dans les secteurs suivants :

- Les grandes entreprises en association avec la gestion des connaissances (en anglais Knowledge Management): pour former et actualiser les connaissances de ses employés de manière plus rapide qu'à l'ordinaire.
- Les universités : Offrent des alternatives de formation à distance. Universités virtuelles qui donnent accès à l'éducation autrement pas possible pour certains étudiants en raison des contraintes de temps ou d'espace ou parce qu'il est trop coûteux.
- Tout particulier qui voudrait entamer un apprentissage en ligne.

3.3. LES COMMUNAUTES D'APPRENTISSAGE

Cette partie a pour objet de donner une idée sur le concept de communauté en essayant de présenter des exemples réels sur quelques communautés. Ensuite, nous montrons la différence entre communauté réelle et communauté virtuelle. Enfin, nous mettons l'accent sur les communautés virtuelles d'apprentissage.

3.3.1. QU'EST CE QU'UNE COMMUNAUTE D'APPRENTISSAGE ?

D'après Dillenbourg dans **[DPC03]**, Une communauté d'apprentissage est un groupe de personnes qui se rassemblent pour acquérir des connaissances. Son objectif est d'augmenter le savoir collectif par l'implication de chaque participant au développement de son savoir individuel **[SB94]**.

Les communautés s'organisent autour d'un espace d'interaction et de partage. Cet espace peut être un lieu physique dans le cas des communautés réelles et un espace virtuel (à travers les TIC et l'Internet) dans le cas des communautés virtuelles. Alors, toutes les communautés utilisent un même espace ou bien la même infrastructure et chaque membre d'une communauté peut interagir avec n'importe quel membre des autres communautés ou de la même communauté où il appartient. Plusieurs communications ou interactions peuvent se faire en même temps entre les différents membres ou groupes. Les interactions au sein de ces communautés incluent par exemple; la mise sur pied d'activités collectives entre les classes d'étudiants des différents enseignants.

3.3.2. LES COMMUNAUTES VIRTUELLES

La communauté virtuelle est un groupe de personnes qui se rencontrent (discutent, échangent, travaillent) sur Internet au moyen des fonctionnalités offertes par le réseau (site web, mail, forum, groupware, blog, wiki...). Les espaces de rencontre sont déterritorialisés et les moyens utilisés sont numériques. Une communauté virtuelle est une forme particulière de réseau

social : elle n’est pas fondée sur les relations interpersonnelles (les participants s’adressent à l’ensemble) **[Gon08]**.

3.3.3. TYPES DE COMMUNAUTES VIRTUELLES

Il existe plusieurs types de communautés, ces communautés ont le même principe mais ils différent par la nature des objectifs à atteindre et l’orientation de leurs activités. Nous citons ici quelques exemples des communautés les plus existantes.

3.3.3.1. Communauté d’intérêt

On parle de « communauté virtuelle d’intérêt » pour un groupe de personnes réunies autour d’un thème d’intérêt commun plus ou moins général (la santé, les enfants, la cuisine, parfois aussi une marque, un produit…) **[Gon08]**.

Une communauté d'intérêt est un groupe composé d'individus qui partagent soit une identité, soit des expériences et des préoccupations. Elle se compose de personnes qui sont personnellement touchées par un problème commun, soit directement, soit dans leur entourage. L'appartenance à une communauté de ce type les aide à comprendre **[Cdi09]**.

3.3.3.2. Communauté de pratique

Une communauté de pratique est un groupe des employés d’une même organisation ou de plusieurs organisations qui collaborent en dehors des cadres établis par leur organisation **[DPC03]**. On parle de communauté virtuelle de pratique (CoPs) lorsque le sujet commun est lié à une pratique professionnelle **[Gon08]**. Ces individus interagissent sur une base continue en vue de maîtriser et d’améliorer les savoirs et savoir-faire de leur domaine d’intérêt. Ainsi, la participation par qui l’apprentissage se déploie, demeure un élément moteur dans une communauté de pratique et revêt un double sens d’implication et d’engagement **[Cda09]**.

Une communauté de pratique virtuelle est définie aussi comme un réseau de personnes partageant un centre d’intérêt autour duquel elles communiquent en ligne. Les praticiens partagent des ressources, notamment des expériences, des problèmes et leurs solutions, des outils et des méthodes. Cette communication permet d’approfondir les connaissances de chaque participant à la communauté et contribue au développement du savoir dans le domaine en question **[Cdp09]**.

3.3.3.3. Communauté d’apprentissage

En revanche, une communauté d’apprentissage virtuelle, même si elle peut inclure la réalisation de recherches originales, a en général pour objectif principal d’accroître le savoir des participants, par le biais de l’éducation formelle ou du perfectionnement professionnel. Dans les

communautés d’apprentissage virtuelles, l’apprentissage peut être soit l’objectif principal, soit un effet secondaire sous la forme de l’E-learning **[Cdp09]**.

3.3.4. COMMUNAUTES VIRTUELLES D’APPRENTISSAGE

Les communautés virtuelles d’apprentissage sont réelles ; elles comprennent de vraies personnes, des enjeux importants et de nombreuses ressources. Le mot virtuel ici ne caractérise pas la communauté elle même mais un de ses modes de communication **[DPC03]**. Elles sont similaires aux «communautés de pratique» mais ne sont pas basées autour d'une profession ou un ensemble de pratiques. Plutôt, elles sont basées sur le désir des participants de la communauté à apprendre les uns des autres **[Hca09]**.

Il s’agit de toute communauté virtuelle réunie dans le but d’apprendre ou, institutionnellement parlant, composée de formateurs et d’apprenants. Dans la communauté, chacun apporte (et doit être autorisé à apporter) ses compétences et les transmet. Les formateurs n'en gardent pas moins leur légitimité, ne serait-ce qu’au titre de médiateur **[Gon08]**. Les communautés varient en :

- Taille : un nombre différent d’enseignants, d’experts, de formateurs, etc.
- Dispersion géographique : les communautés peuvent mettre en relation soit des enseignants des régions et/ou pays différents ou bien qui appartiennent à la même région et/ou pays.
- Durée de vie : en effet, la durée de vie des communautés locales est plus longue que celle des communautés internationales qui sont souvent liées à un projet et survivent rarement au-delà du terme de ce projet.

3.4. TRAVAIL COLLABORATIF

La collaboration constitue un mot clé pour le présent mémoire par conséquent il serait évident de consacrer une partie pour la décrire. Dans ce qui suit nous allons présenter un panorama rapide des concepts liés à la collaboration notamment travail collaboratif, apprentissage collaboratif et espace collaboratif.

3.4.1. CSCW

Le CSCW (Computer Supported Cooperative Work), ou en français TCAO (Travail Coopératif Assisté par Ordinateur) désigne les Systèmes informatiques qui supportent des groupes de gens engagés dans une tâche (ou un but) commune, et qui fournissent une interface sur un environnement partagé **[Par05]**.

Trois types d'activités doivent être supportés par le CSCW **[Par05]**.

- la co-production, désigne la réalisation d'un travail en commun,
- la communication, qui permet aux utilisateurs de synchroniser leur réalisation,
- la coordination, qui concerne l'organisation formelle du travail.

3.4.2. CSCL

CSCL (Computer Supported Collaborative Learning) est considéré comme un nouveau paradigme pour l'apprentissage collaboratif. Il permet aux individus qui sont éloignés de collaborer. Le CSCL est une méthode qui apporte les avantages de l'apprentissage collaboratif et l'apprentissage coopératif aux utilisateurs qui suivent un apprentissage distant via un réseau d'ordinateurs **[Csc09]**.

L'objectif du CSCL est d'aider les utilisateurs d'apprendre ensemble de manière efficace. Il soutient la communication d'idées et d'informations entre les apprenants et l'accès collaboratif aux informations et aux documents.

3.4.3. QU'EST-CE QUE L'APPRENTISSAGE COLLABORATIF?

Marc Walckiers et Thomas De Praetere dans **[WP04]**, ont proposé la définition suivante :

L'apprentissage collaboratif est toute activité d'apprentissage réalisée par un groupe d'apprenants ayant un but commun, étant chacun source d'information, de motivation, d'interaction, d'entraide… et bénéficiant chacun des apports des autres, de la synergie du groupe et de l'aide d'un formateur facilitant les apprentissages individuels et collectifs.

L'apprentissage collaboratif est une démarche active par laquelle l'apprenant travaille à la construction de ses connaissances. Le formateur y joue le rôle de facilitateur des apprentissages alors que le groupe y participe comme source d'information, comme agent de motivation, comme moyen d'entraide et de soutien mutuel et comme lieu privilégié d'interaction pour la construction collective des connaissances **[FL01]**.

3.4.4. ESPACE COLLABORATIF

Est une composition de plusieurs services dont l'objet est de gérer des entités (groupes, membres, droits, services) ainsi que les interactions et les flux d'information entre ces entités **[Dug07]**.

3.4.5. COLLABORATION ET COOPERATION

Les deux termes collaboration et coopération sont des termes qui se ressemblent et dont la distinction entre eux n'est pas toujours facile. La littérature définit la collaboration comme un

effort mutuel et coordonné des participants pour résoudre ensemble un problème, tandis que la coopération correspondrait plutôt à une organisation partagée du travail dans laquelle chacun est responsable d'une portion de problème à résoudre **[DBBO95]**.

3.5. POSITIONNEMENT DE NOTRE PROPOSITION

Dans les sections précédentes de ce chapitre nous avons montré les concepts fondamentaux liés au E-learning, communautés virtuelles d'apprentissage, et une brève présentation de l'apprentissage collaboratif, la coopération et la collaboration. Après avoir présenté dans le premier chapitre la technologie de Grid et ses applications, nous pouvons déduire quels avantages peuvent les Grids apporter aux environnements de collaboration et aux systèmes de E-learning. Par conséquent, nous constatons indispensable de consacrer cette partie afin de montrer l'utilité d'employer le Grid pour la collaboration dans le domaine d'éducation et d'apprentissage. Pour cela, nous allons d'abord présenter les inconvénients des solutions de E-learning actuelles, ensuite nous montrons l'apport de Grid pour le E-learning et pour les environnements de collaboration.

3.5.1. LES INCONVENIENTS DES SOLUTIONS ACTUELLES DE E-LEARNING

Selon Capuano dans **[CGL+03]**, l'apprentissage en ligne a beaucoup d'avantages par rapport aux modèles d'apprentissage traditionnels:

- Une meilleure interaction entre les apprenants et les ressources d'apprentissage qu'ils utilisent, c.-à-d. l'apprentissage n'est pas passif,
- L'apprentissage peut se produire n'importe où et à n'importe quel moment ; il n'y a pas de frontières, liées au temps et de lieu,
- Les tuteurs, ou les apprenants eux-mêmes sont capables de suivre les progrès et de personnaliser l'expérience d'apprentissage s'appuyant sur les compétences de l'apprenant et ses préférences.

Cependant, les solutions actuelles présentent plusieurs inconvénients **[LTT08]** :

- Elles sont principalement centrées sur la livraison de contenu,
- Les plateformes d'apprentissage actuelles prennent uniquement en charge un domaine d'apprentissage spécifique et ne sont pas en mesure de soutenir l'apprentissage dans différents domaines.
- De nombreuses plates-formes et systèmes de E-learning ont été développés et commercialisés, mais avec des limitations en termes d'évolutivité, de disponibilité et de distribution de puissance de calcul ainsi que des capacités de stockage.

3.5.2. APPORT DE GRID POUR LE E-LEARNING

La technologie de Grid est devenue très populaire dans l'éducation grâce aux avantages qu'elle offre, elle représente le contexte idéal pour l'éducation. Parmi ses avantages nous citons :

- L'évolutivité à un large nombre d'apprenants ou de groupe d'apprenants et des tuteurs qui sont géographiquement distribués.
- La disponibilité d'un nombre croissant de ressources d'apprentissage hétérogènes et évolutives.
- L'intégration des ressources d'apprentissage existantes et de nouvelles ressources et outils, ainsi que d'autres systèmes appartenant à différentes organisations.
- le support des notifications aux participants sur ce qui se passe dans leur environnement.
- l'interopérabilité en fournissant un support pour une variété des plateformes et logiciels des apprenants et tuteurs.

Tandis que le Grid est souvent considéré comme une infrastructure qui fournit un système distribué des ressources de calcul de haute performance, ce n'est là qu'un aspect de l'utilisation réussie de Grid de calcul parmi plusieurs. Nous pouvons considérer les ressources de calcul distribuées comme des services fournis par plusieurs organisations, qui sont utilisés pour enrichir et améliorer l'environnement de E-learning. Les technologies de Grid peuvent satisfaire les besoins d'un environnement d'apprentissage efficace : gestion des ressources dynamiques et des utilisateurs, partage de ressources, intégration et découverte, et la sécurité à travers des domaines administratifs différents et multi-institutinnels **[JP07]**.

3.5.3. APPORT DE GRID POUR LES ENVIRONNEMENTS DE COLLABORATION

Comment le Grid peut-il faciliter le développement des technologies de collaboration ? **[LYJS06]**.

Compte tenu de l'influence des services Web dans les standards de Grid émergents, il est évident que la capacité de partage coordonné de ressources par le Grid va devenir plus ubiquitaire, ce qui est très important pour la poursuite du développement de CSCW, en particulier pour la réalisation d'une collaboration à grande échelle.

Deuxièmement, dans un environnement de Grid, une organisation virtuelle (OV) est constituée d'un ensemble d'individus/organisations distribués et hétérogènes fournissant des ressources/services, dont chacune doit faire face à la diversité des technologies d'accès parmi ses participants. L'OV est essentiellement façonnée par l'infrastructures de sécurité de Grid GSI, parce que les aspects et les mécanismes de sécurité dans un environnement de Grid vont bien au-

delà de l'établissement traditionnel de relations de confiance entre un client et un serveur, comme le mécanisme d'authentification unique SSO fourni par l'infrastructure GSI, selon lequel une seule action d'authentification des utilisateurs et d'autorisation peut permettre à un utilisateur d'accéder à tous les ordinateurs et systèmes où il a l'autorisation d'accès sans qu'il soit nécessaire d'entrer les mots de passe multiples. La gestion des OVs et les services de sécurité de Grid sont des caractéristiques importantes pour la collaboration distribuée à grande échelle car la communication ouverte et libre nécessite des mécanismes pour connaître les identités des participants (e.g., identités personnelles, identités des groupes), qui peuvent être facilement implémentés dans un environnement de Grid. Les Utilisateurs dans les OV sont organisés en groupes qui forment en général une structure hiérarchique avec l'OV elle-même comme racine et les mécanismes de la création/maintien de la structure de collaboration peuvent être fournis pour promouvoir la coopération centrée sur le groupe. En outre, l'environnement de collaboration sécurisé peut être conçu pour soutenir la collaboration informelle spontanée ainsi que la collaboration hautement formelle.

Troisièmement, le modèle de Grid offre un cadre de base pour la création des environnements coopératifs ; par exemple, OGSA fournit un ensemble de services de base qui peut être étendu pour la mise en œuvre de services de haut niveau, y compris la découverte de services, la création/suppression dynamique de services, la gestion de vie, etc.

En résumé, le Grid dispose des caractéristiques et des fonctionnalités qui peuvent répondre aux exigences des CSCW et CSCL et par conséquent, le Grid peut servir pour bâtir de nouveaux environnements de collaboration.

3.6. SYNTHESE DE QUELQUES TRAVAUX

Cette section est réservée pour la présentation de quelques travaux parmi plusieurs qui se basent sur le Grid et bénéficient de ses potentialités, et ceci dans le cadre du E-learning et de collaboration. En réalité, Il existe d'autres travaux dont la plupart forment des solutions très complexes et qui sont la récolte de plusieurs travaux et recherches. Par conséquent, pour ne pas nous éloigner de notre objectif nous avons sélectionné ceux qui sont plus proches de notre préoccupation.

3.6.1. PRESENTATION DES TRAVAUX

3.6.1.1. Grid de E-learning

L'objectif de ce travail qui est appelé "*E-learning Grid*" **[PV03]** (en français Grid de E-learning) est d'utiliser le Grid pour fournir une grille d'apprentissage en introduisant dans les

objets d'apprentissage les applications gourmandes en calcul qui traitent de grandes bases de données tels que la visualisation photo-réaliste, et les simulations complexes en temps réel. La raison pour l'ignorance de ce genre d'application est les problèmes de faisabilité technique et les coûts élevés.

Dans les universités, souvent, l'infrastructure matérielle existe et peut être formée d'une centaine de PCs et même de supercalculateurs. Cependant, le taux d'utilisation de cette infrastructure est très bas. L'objectif de ce travail, en effet, est de créer un Grid pour le E-learning "*E-learning Grid*" dans lequel les fonctionnalités de Grid de calcul sont intégrées dans les systèmes de E-learning.

Dans **[PV03]** Pankratius et Vossen proposent une architecture d'une grille d'apprentissage qui se compose de deux parties principales : un LMS qui est basé sur les web services et l'intergiciel de Grid. Ils introduisent également le concept de GLOB (Grid Learning OBjects) qui comprend des ressources d'apprentissage classiques ainsi que d'une couche d'application pour l'utilisation des services de Grid. La figure 3.2 est une représentation simplifiée de leur concept sans indiquer en détail les services internes.

Lors de la connexion initiale au LMS, la couche fabrique de Grid est fournie au dispositif de l'apprenant sous forme d'une applet Java. Le service Login du LMS est en mesure d'authentifier l'utilisateur et son dispositif contre le Grid et on peut également lui faire confiance pour créer ou supprimer des utilisateurs du réseau. Le LMS demandera un TGT (ticket-granting-ticket) de l'intergiciel de Grid et l'apporte à son client. Si le GLOB est délivré au dispositif de l'apprenant, il est capable d'utiliser la couche fabrique et le TGT initialement fourni par le LMS pour demander d'autres billets de session et accéder directement aux services de Grid.

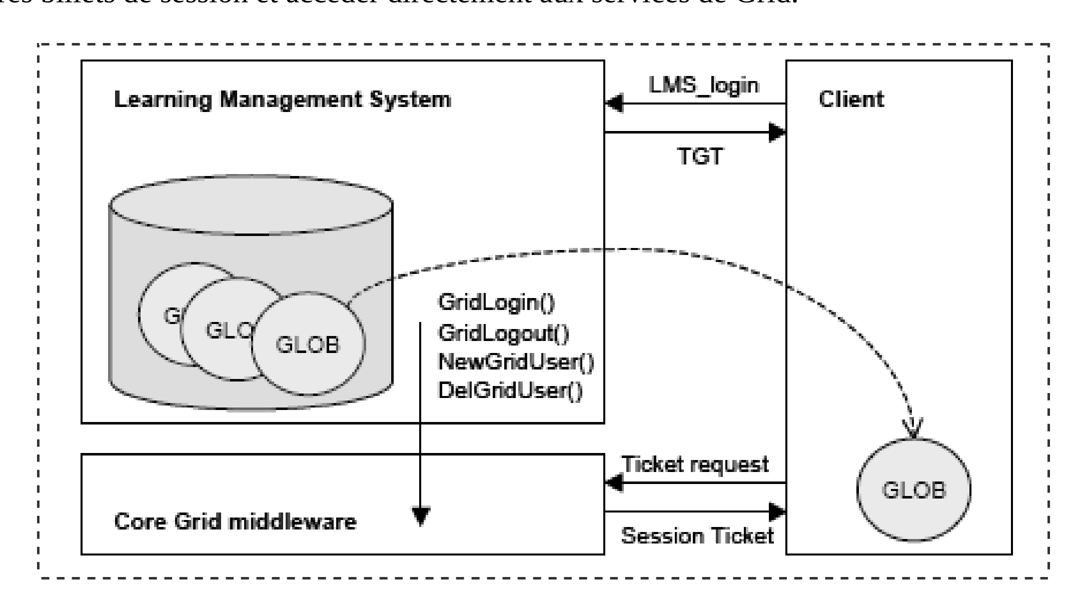


Figure 3. 2 Architecture de 'Learning Grid' (source [PV03])

L'architecture proposée fournit un couplage faible entre le LMS et l'intergiciel de Grid avec quelques exigences pour le LMS, tel que l'appel de quelques opérations par le *service login* et la fourniture de l'applet au client au moment de l'authentification. Il semble donc possible d'améliorer presque tous les LMS basés sur le web de cette manière, indépendamment du fait que le LMS est basé sur les web services ou non. Parfois, cependant, il ne sera pas possible d'introduire le Grid par extension de l'LMS.

3.6.1.2. Grid-cluster

Les plateformes de E-learning généralement nécessitent des dispositifs de grande capacité de stockage, afin de stocker le contenu d'apprentissage. Cependant, les écoles à faible budget ne peuvent pas fournir ces équipements qui sont chers et coûteux, et le matériel qu'elles détiennent n'est pas pleinement utilisé. L'idée est alors d'exploiter les ressources sous utilisées en connectant ces ordinateurs à l'aide de la technologie de Grid de calcul, par conséquent l'espace de stockage obtenu est égal à celui des dispositifs de stockage qui sont chers. La connexion des dispositifs de stockage sous utilisés se fait par les techniques de Data Grid.

Cette architecture est appelée Grid-cluster **[YH05]**, implémentée au dessus de Globus Toolkit. Elle est composée de trois clusters, qui présentent chacun un nœud de quatre PCs : un maître et trois esclaves. Les nœuds sont connectés, via une carte Ethernet, à un Switch HUB. Elle utilise un portail de Grid qui fournit un ensemble de services pour les membres et joue le rôle d'une plateforme d'apprentissage pour les autres institutions via internet. Cet environnement utilise le portail *Open Grid Computing Environment (OGCE)*. L'architecture de cette plateforme est présentée dans la figure 3.3 ci-dessous :

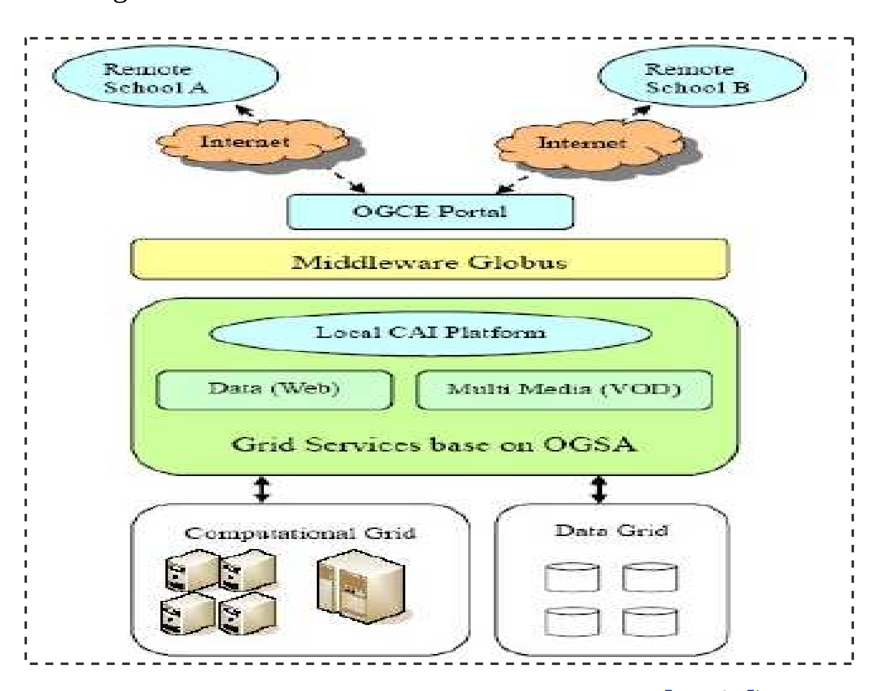


Figure 3. 3 Architecture de Grid-cluster (source [YH05])

3.6.1.3. Architecture de E-learning basée sur le Grid

L'objet de ce travail **[Che08, LTT08]** est la proposition d'une architecture pour le E-learning basée sur le Grid. Ce travail vise l'apprentissage à distance qui est un des modes d'apprentissage. En effet, le problème qui se pose est celui lié à la manière de soutenir les systèmes d'apprentissage existants dans des environnements dynamiques, extensibles et hétérogènes. Le scénario est un environnement informatique à grande échelle qui interconnecte des LMSs, des LCMSs et des systèmes de classes virtuelles de différentes organisations.

Ce travail résulte en une architecture collaborative de l'apprentissage à distance basée sur la technologie de Grid dans un environnement hétérogène. En combinant la technologie de Grid et celle de l'apprentissage à distance il est possible de bâtir un système d'apprentissage ubiquitaire et efficace en donnant la possibilité de partage de ressources d'apprentissage dans des environnements hétérogènes et distribués. Les apprenants peuvent à n'importe quel moment accéder à un cours de leur choix à partir d'un répertoire virtuel de contenu.

Cette architecture contient cinq couches du bas en haut comme le montre la figure 3.4.

i. *La couche infrastructure*, dans le niveau le plus bas, contient l'environnement réseau de base, incluant les dispositifs de calcul, les protocoles réseau, etc.
ii. *La seconde couche*, architecture orientée service pour l'implémentation des services Web de base reliés aux protocoles tel que XML, UDDI/SOAP/WSDL. Cette couche fournit la flexibilité, la fiabilité et la connectivité élémentaire pour les couches supérieures.
iii. *La couche du middleware de Grid* est le cœur de cette architecture où les problèmes de base de Grid sont résolus. Le middleware utilisé est Globus Toolkit 4 (GT4), cette boite à outils fournit les quatre services de base; services de sécurité, gestion de ressources, gestion de données et les services d'information.
iv. *La couche du contenu* est au dessus de la couche middleware qui sert à stocker tous les contenus d'apprentissage dans la plateforme avec l'utilisation du format SCORM pour rendre les cours réutilisables et acceptables aux différents LMSs.
v. Enfin, *le portail* de la grille d'apprentissage est l'entrée unifiée de tous les utilisateurs de la plateforme de Grid, soutient l'authentification unique SSO (Single Sign On) des utilisateurs sur le système. Le portail permet aux utilisateurs le partage de ressources d'apprentissage sans savoir d'où proviennent elles.

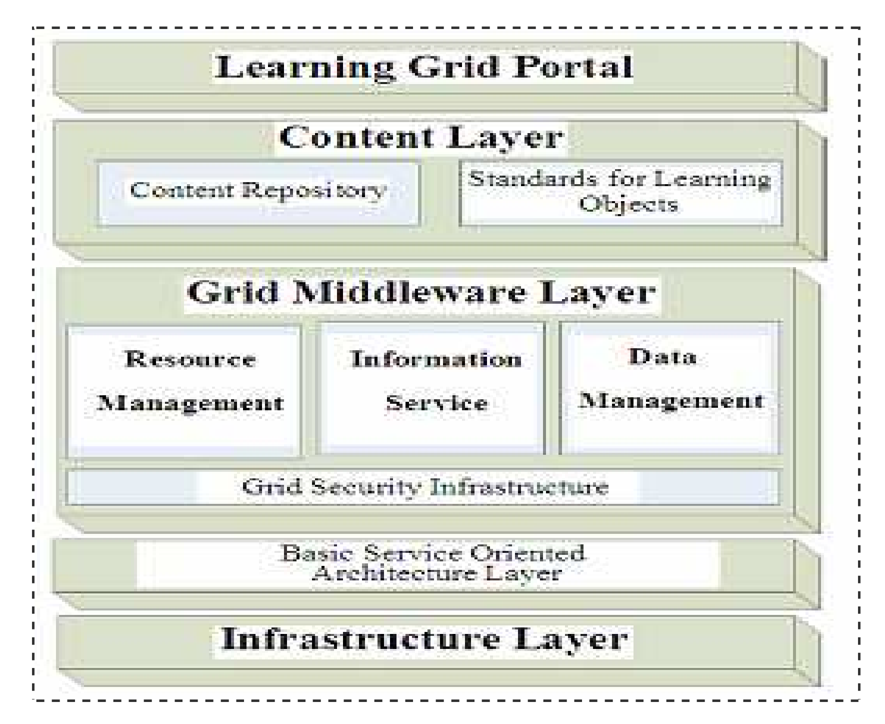


Figure 3. 4 Architecture de Grid de E-learning (source [Che08])

3.6.1.4. GESOTC (Grid Environment for Sharing Objects)

L'objectif de ce travail **[BTNK07]** est l'implémentation d'un environnement de collaboration pour les communautés à centre d'intérêt commun dans le domaine de formation, qui sont souvent hétérogènes. Cet environnement est basé sur l'infrastructure GLS (Grid Learning Services), et a pour objectif le regroupement des communautés, offrir aux communautés peu performantes l'opportunité de s'améliorer, et l'harmonisation des systèmes de formation. Cet environnement est nommé GESOTC (Grid Environment for Sharing Objets).

GESOTC est utilisé au dessus des plateformes de collaboration telle que DyColo (Dynamic Collaboration Learning Object). La plateforme DyColo tient compte de la collaboration à l'intérieur d'une communauté en permettant aux différents membres d'une communauté de se regrouper et de partager les ressources de formation. Il est basé sur le GLS qui fournit plusieurs services tels que les services de e-qualification, de communication et de collaboration. Le rôle de GESTOC est alors d'apporter une solution au problème de la collaboration entre plusieurs communautés de formation en exploitant les services offerts par le GLS.

GESOTC a deux grands objectifs distincts, la collaboration au niveau macroscopique (collaboration entre communautés) et la collaboration microscopique (au niveau interne à la communauté). La figure 3.5 montre le fonctionnement de GESOTC. Il est essentiellement composé des communautés d'échange, des utilisateurs de ces différentes communautés et des ressources à mettre à disposition.

A l'aide de l'interface fournie par GESOTC la communauté peut s'inscrire, par conséquent, cette dernière va bénéficier de tous les outils macroscopiques qui sont offerts par l'environnement et de même, les membres vont utiliser les services microscopiques offerts par le GLS. Un utilisateur donné, selon sa classification sera affecté à la communauté virtuelle appropriée. Une fois les membres de la communauté sont reconnus, ils peuvent collaborer ensemble selon leur qualification. La collaboration peut aussi apparaître entre les gestionnaires de communautés, les formateurs et les apprenants. En conclusion, GESOTC offre une possibilité de collaboration aux utilisateurs à l'intérieur d'une communauté et une collaboration à l'extérieur de la communauté avec les membres qualifiés.

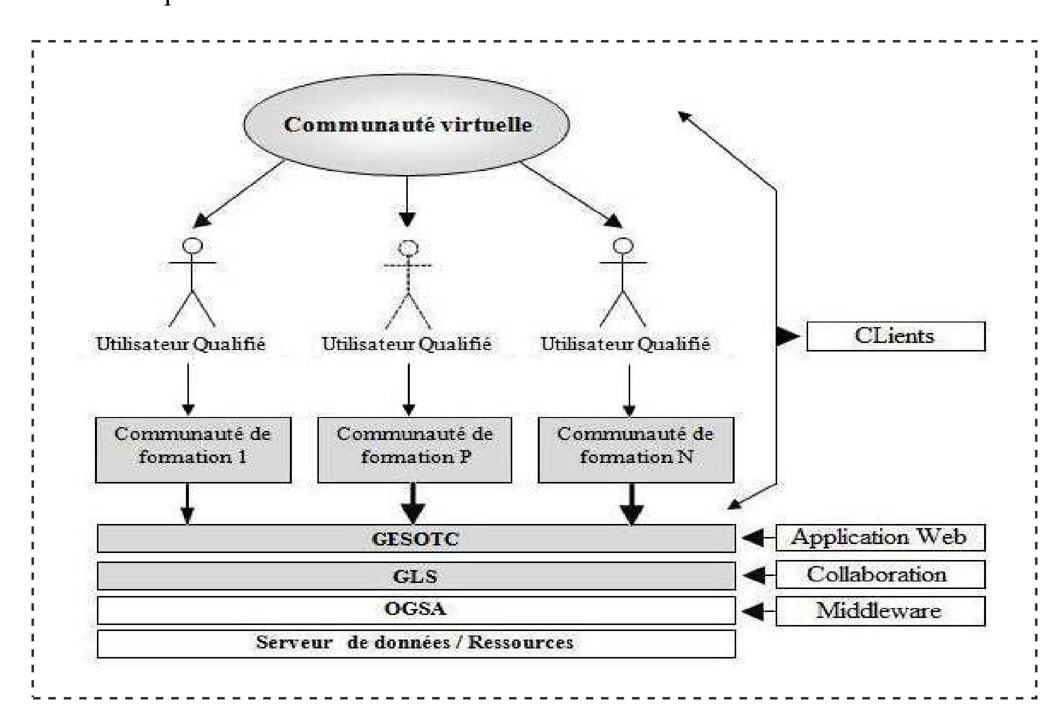


Figure 3. 5 Schéma général de fonctionnement (source [BTNK07])

La figure 3.5 ci-dessus définit une structure de Grid composée d'une partie centrale (OGSA) au dessus de laquelle se trouve une couche de services de collaboration (GLS) et établit une interaction avec un LMS (GESOTC). Il est composé de trois couches.

i. *Une couche Grid Middleware* : le middleware est l'élément central de Grid, fournit plusieurs services à savoir, la communication, le transfert de données, la sécurité, etc. il est composé des couches de l'architecture OGSA.
ii. *Une couche GLS* : basée sur OGSA et constituée des services essentiels dans le domaine de l'éducation, tels que les services de e-qualification, de collaboration et de communication. C'est sur cette couche que l'environnement GESTOC est bâti.

iii. *Une couche LMS* : c'est le cœur de GESTOC. L'accès à l'environnement se fait à l'aide d'un navigateur, cet accès va donner aux utilisateurs l'accès aussi aux services et ressources de Grid.

Ce travail traite le problème de collaboration entre les communautés de formation qui partagent les mêmes objectifs. Cet environnement est basé sur la couche GLS qui elle même est basée sur l'infrastructure de Grid. Le GLS fournit pour l'environnement GESOTC un ensemble de services de collaboration qui peuvent être utilisée à l'intérieur d'une communauté. Le rôle de GESOTC est alors de fournir une collaboration entre plusieurs communautés en exploitant les services de GLS et d'améliorer la communication et l'interaction entre les membres des différentes communautés de formation en mettant à leur disposition un panel ouvert de services de collaboration. Les gestionnaires de communautés auront la possibilité de partager leurs expériences et d'évaluer leur communauté de formation ; ils peuvent combler leurs besoins en ressources pédagogiques en sollicitant l'aide des autres communautés. Les formateurs peuvent harmoniser leur enseignement à l'issue des évaluations faites en commun dans le réseau d'échange, et les apprenants peuvent collaborer entre eux suivant leur niveau d'étude.

3.6.2. BILAN

Dans cette partie, nous avons exposé quelques systèmes de E-learning qui se basent sur le Grid. Nous remarquons ici que chacune des architectures précitées utilise le Grid pour résoudre un problème donné. Le travail de Pankratius **[PV03]** présente une architecture d'une grille d'apprentissage qui se compose de deux parties principales : un LMS qui est basé sur les Web services et l'intergiciel de Grid. Il introduit également dans les objets d'apprentissage un nouveau type de contenu qui est les applications gourmandes en calcul tel que la simulation et la visualisation, l'invocation de ce genre d'applications lors de l'apprentissage implique leur exécution sur les ressources de Grid. Le travail présenté par YANG dans **[YH05]** joue le rôle d'une plateforme d'apprentissage pour diverses institutions via Internet. Il utilise le Grid pour connecter les ordinateurs appartenants aux écoles et institutions, et les techniques de Data Grid pour relier les dispositifs de stockage. L'entrée unifiée de cette plateforme est un portail basé sur OGCE. Chen et Li **[Che08, LTT08]** visent l'enseignement ou l'apprentissage à distance et traitent le problème de partage de contenu qui est délivré par une variété de LMS et LCMS et qui est par conséquent hétérogène. Ils utilisent, pour permettre le partage de contenu, les formats SCORM. Finalement, Batchakui dans **[BTNK07]** traite le problème de collaboration à l'intérieur et à l'extérieur d'une communauté de formation. GESOTC représente une plateforme d'apprentissage et de collaboration et chaque utilisateur de la plateforme (apprenant) est affecté à la communauté

la plus appropriée selon son niveau et ses préférences, cela se fait à l'aide de la couche GLS qui est basée sur le Grid sémantique.

Nous constatons à partir de ces travaux la variété des modes d'usage de Grid dans le domaine d'éducation et d'apprentissage et comment la complexité de l'architecture augmente avec l'augmentation et l'évolution des objectifs à atteindre. Le Grid de calcul est largement exploité dans ce domaine, mais l'appel à de nouveaux concepts et techniques s'avère indispensable.

Le tableau 2.3 ci-dessous présente un récapitulatif sur les travaux que nous avons présentés, et résume les principaux outils et technologies employés.

Tableau 2. 3 Comparaison entre quelques environnements d'apprentissage à base de Grid

	Objectif	Architecture	Utilisation d'un LMS	Utilisation de portail	Utilisation d'une ontologie	Utilisation des techniques de Data Grid	Middleware Globus
E-learning Grid" [PV03]	-grille d'apprentissage -Concept de GLOB	SOA[5]	X				X
Grid-cluster [YH05]	-grille d'apprentissage -Exploiter les ressources sous utilisées des écoles en connectant ces ordinateurs	SOA		X		X	X
Architecture de E-learning basée sur le Grid [Che08, LTT08]	-soutenir les systèmes d'apprentissage existants dans des environnements dynamiques, extensibles et hétérogènes -partage de contenu d'apprentissage	SOA		X			X
GESOTC [BTNK07]	-Environnement de collaboration pour des communautés de formation à centre d'intérêt commun -Soutient la collaboration microscopique et macroscopique	SOA	X	X	X		X

[5] Le terme SOA désigne ici architecture orientée service c'est l'équivalent en anglais de "Service Oriented Architecture".

Les travaux exposés présentent plusieurs avantages et répondent bien aux objectifs fixés et la majorité de ces travaux n'ont pas été implémentés. Cependant, aucun de ces systèmes ne fournit une possibilité pour la collaboration des communautés virtuelles utilisant des LMSs différents c.-à-d. soit qu'ils offrent une solution complète de E-learning sans essayer d'utiliser celles existantes, soit qu'ils utilisent les LMSs existants et permettent le partage de contenu mais sans donner un support pour la collaboration avec tout ce qu'elle peut inclure comme services et outils. Pour cette raison, notre objectif est de proposer une architecture d'un environnement de collaboration pour les communautés d'apprentissage utilisant des LMSs différents permettant le partage de ressources de différentes natures, la communication et la collaboration entre les participants, ceci via l'utilisation de Grid comme infrastructure de base.

3.7. CONCLUSION

Dans la première partie de ce chapitre nous avons défini les concepts clés d'un système de E-learning, ses objectifs, ses composants, ses acteurs et son domaine d'application. Ensuite, nous avons introduit un deuxième concept qui est celui de 'communauté virtuelle d'apprentissage' (appelée aussi VLC) considérée aujourd'hui comme une nouvelle forme d'apprentissage. La section consacrée aux VLCs a fait appel à une nouvelle et importante section qui est celle du CSCW et CSCL, où nous avons présenté les concepts de base du travail et apprentissage collaboratifs.

La dernière partie est une justification de notre problématique et contribution où nous avons montré en premier lieu les besoins, lacunes des environnements de E-learning et de collaboration et comment le Grid peut leur apporter une solution. Cette partie a été finalisée par une synthèse des travaux existants qui sont en effet des solutions de E-learning à base de Grid.

A partir des différentes parties traitées dans ce chapitre, nous avons compris le lien existant entre les trois concepts qui font l'objet de notre recherche notamment E-learning, collaboration et communautés d'apprentissage qui présentent des mots clés pour ce mémoire et comment le Grid peut apporter une solution pour chacun de ces concepts de manière individuelle. Nous avons aussi montré à travers la section synthèse des travaux les différents modes d'usage de Grid dans le E-learning qui varient avec la variation des objectifs à atteindre. Nous constatons aussi à travers la partie "état de l'art" que le Grid constitue la bonne solution pour le domaine d'éducation et par conséquent un bon outil pour les environnements collaboratifs, mais le Grid tout seul ne peut pas tout résoudre.

Notre objectif qui fera l'objet du chapitre suivant, est de proposer un environnement de collaboration à base de Grid pour les communautés utilisant des LMSs différents en essayant d'utiliser de nouveaux mécanismes et même ceux existants pour renforcer la collaboration entre ces différentes communautés.

PARTIE 2

Contribution

Afin de répondre à notre problématique, qui s'agit d'exploiter les puissances de Grid pour répondre au problème de collaboration ; qui est un problème posé dans de nombreux domaines et plus précisément le domaine de E-learning, nous consacrons ce chapitre pour la proposition d'une architecture qui facilite la collaboration entre différentes communautés qui possède chacune son propre LMS.

CHAPITRE 4

ARCHITECTURE PROPOSEE

" C'est avec la logique que nous prouvons et avec l'intuition que nous trouvons"
Henri Poincare.

Dans ce chapitre, nous présentons notre architecture qui s'agit d'un environnement de collaboration pour les communautés d'apprentissage, qui peuvent être formées d'universités et d'institutions qui sont géographiquement distribuées. Le but de ce chapitre est de montrer en premier lieu quelles sont les exigences d'un environnement de collaboration pour l'apprentissage selon notre vision et selon ce qui a été présenté dans la littérature, en second lieu, essayer d'interpréter ces exigences et besoins sous forme d'une architecture qui va apporter une solution au problème de collaboration entre plusieurs communautés distantes, et qui résout la plupart des problèmes qui peuvent être rencontrés, tels que l'héterogenieté et la dispersion géographique des communautés, le partage coordonné des ressources, la sécurité et la gestion des autorisations. Pour ce faire, nous adoptons le Grid comme infrastructure de base pour notre environnement, qui apporte des améliorations significatives et offre une solution pour de nombreux problèmes.

4.1. INTRODUCTION

La technologie de Grid a été initialement conçue pour fournir un support aux applications appartenant au domaine scientifique. Cependant, aujourd'hui avec l'apparition du standard OGSA le Grid a touché plusieurs domaines tels que le commerce et l'éducation. A l'heure actuelle, l'éducation et la collaboration sont considérées comme les applications les plus importantes de Grid. L'utilisation de Grid permet de donner accès à de larges quantités de ressources hétérogènes qui peuvent être utilisées à des fins éducatives.

La collaboration est considérée comme un des points les plus importants dans un système de E-learning, elle permet l'échange de connaissances, de compétences et d'idées sur un sujet particulier en donnant aux différents acteurs (apprenants, enseignants, administrateurs, ...) l'occasion de travailler ensemble pour la création de cours, évaluation des résultats, etc. Ainsi, il existe un besoin croissant de fournir et de concevoir un environnement de collaboration dans le domaine du E-learning, non seulement pour améliorer l'apprentissage collaboratif entre les apprenants, mais aussi pour favoriser la collaboration entre les différentes entités d'une communauté virtuelle d'apprentissage tel que : donner aux communautés la possibilité de travailler ensemble en partageant les ressources et les enseignants et en organisant des séances d'évaluation commune pour les classes similaires. Dans cette proposition nous allons présenter une architecture d'un environnement collaboratif pour le E-learning en se basant sur le Grid, cet environnement est baptisé **GCEL** (**G**rid based **C**ollaborative **E**nvironment for **L**earning).

GCEL fournit une collaboration globale pas uniquement à travers le partage global de ressources, mais en fournissant un environnement de travail partagé qui va permettre aux utilisateurs de collaborer en temps réel. Cet environnement n'est pas dédié pour l'apprentissage collaboratif entre les apprenants d'une même communauté, mais c'est un lieu de rencontre de plusieurs communautés. GCEL va gérer alors tous les aspects liés à la collaboration comme la communication, la sécurité et la notification.

Nous allons alors présenter une vue globale de l'environnement de collaboration par un modèle en couche en décrivant les différents services qui le composent, ensuite nous allons mettre l'accent sur les services de haut niveau fournis par l'environnement.

4.2. L'OBJECTIF DE GCEL

Avant d'aller plus loin, nous résumons dans cette section les principaux objectifs de notre environnement de collaboration.

- Permettre aux communautés d'apprentissage de collaborer en partageant les ressources (objets d'apprentissage, matériel…) et les enseignants.

- Permettre l'apprentissage collaboratif entre un groupe d'apprenants qui appartiennent à des communautés différentes.
- Assurer aux utilisateurs libres, qui n'appartiennent pas forcément à une communauté d'apprentissage, une énorme quantité de ressources et services pour exécuter leurs applications, accéder aux cours de différents formats (texte, audio, vidéo) et même leur donner la possibilité de communiquer avec les enseignants des différentes communautés.
- Intégration des ordinateurs des écoles et des institutions non reliés pour remplacer les serveurs.
- Donner aux communautés (universités, instituts…) qui possèdent des budgets limités la possibilité de s'améliorer via la collaboration et l'entre-aide avec les autres communautés.
- Mieux exploiter les ressources sous utilisées en les intégrant dans une unité globale à l'aide de l'intergiciel Globus.
- Utiliser les plateformes d'apprentissages (LMSs) existantes qui offrent la plupart des services nécessaires au lieu de concevoir toute une nouvelle solution de E-learning.
- Donner aux communautés le choix d'utiliser la plateforme qui leur convient et plus particulièrement les plateformes en open source.

4.3. DEMARCHE DE CONCEPTION

Dans cette section, nous décrivons de manière rapide et concise les principaux points sur lesquels nous nous sommes basés pour concevoir notre environnement de collaboration. Nous commençons par la description globale des composants de l'environnement, puis nous rentrons dans les détails ; structure des utilisateurs, types d'interactions, niveaux de collaboration et enfin une description des exigences d'un environnement de collaboration.

4.3.1. LES COMPOSANTES DE L'ENVIRONNEMENT

En effet, l'élément de base de notre environnement de collaboration est la communauté virtuelle d'apprentissage. Cependant, pour des raisons de simplicité, nous décomposons essentiellement ce large concept en trois entités de base notamment les communautés, les utilisateurs et les ressources qui forment le concept de communauté virtuelle d'apprentissage lorsqu'elles se rassemblent ensemble. Il est à noter que la communauté virtuelle d'apprentissage désignée ici ne s'agit pas de tout groupe interagissant via Internet tel que défini par quelques auteurs, mais nous désignons le concept d'organisation virtuelle dans le cadre d'apprentissage et

de formation où plusieurs communautés d'apprentissage qui sont géographiquement dispersées collaborent et travaillent ensemble généralement à travers l'Internet et les technologies d'information et de communication. Aussi, lorsque nous disons communauté d'apprentissage cela peut être remplacé par communauté de formation, car ces deux termes ou concepts (communauté d'apprentissage et communauté de formation) sont dépendants et l'un d'entre eux ne peut avoir lieu sans l'existence de l'autre ; nous ne pouvons pas parler d'enseignants sans qu'il y ait des apprenants et vice versa. Tout au long de ce travail, nous suivons la définition de "communauté virtuelle d'apprentissage" suivante donnée par Isabelle Gonon dans son guide pratique **[Gon08]**:

Il s'agit de toute communauté virtuelle réunie dans le but d'apprendre ou, institutionnellement parlant, composée de formateurs et d'apprenants.

4.3.1.1. La communauté

Le type de communauté désigné ici est toute communauté qui a pour objectif l'acquisition et la diffusion de la connaissance. Une communauté peut être formée des établissements, des universités, des institutions, etc. Un caractère commun de ces universités et établissements est leur dispersion géographique ainsi que la nature de la matière enseignée/fournie. Donc, une communauté peut être considérée comme une source de connaissances multiples.

Chaque communauté est composée de :

- Un administrateur : c'est le responsable de la communauté
- Des enseignants : ce sont des enseignants, qui peuvent être soit des auteurs, des formateurs, tels qu'ils sont définis dans le chapitre 3.
- Des experts s'ils existent.

La figure 4.1 ci-dessous montre la catégorie des organisations qui vont utiliser l'environnement de collaboration que nous allons présenter durant ce chapitre.

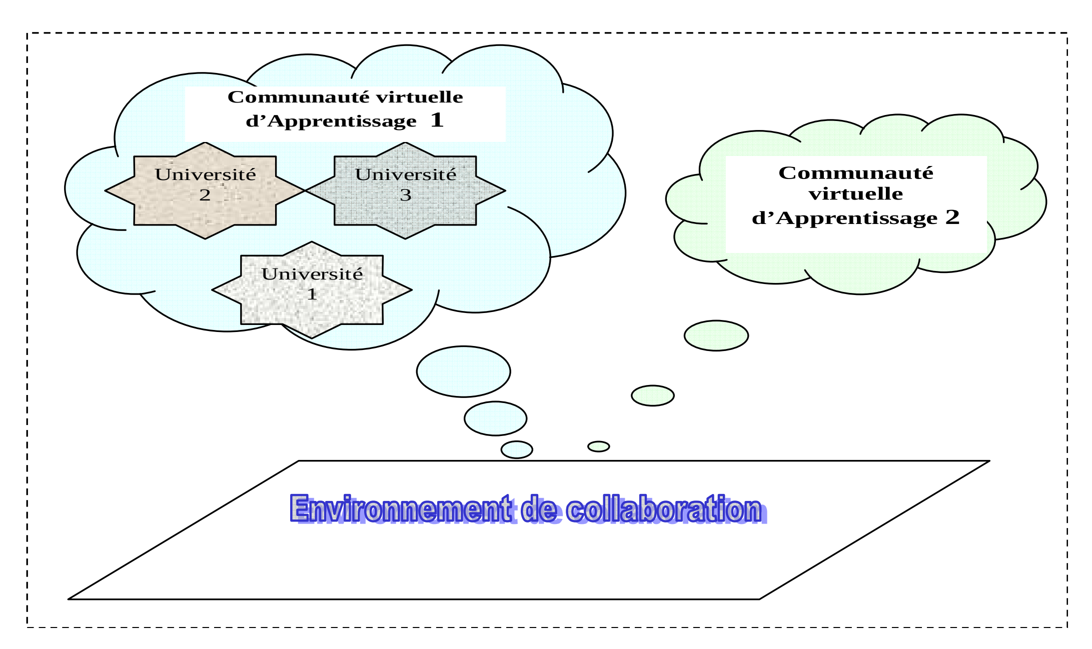


Figure 4. 1 Les entités utilisatrices de l'environnement de collaboration

4.3.1.2. Utilisateurs ou apprenants

C'est tout particulier voulant apprendre ou se former dans un domaine donné, ils forment des entités externes à la communauté. Nous distinguons ici deux types d'utilisateurs :

- **Utilisateur libre** : c'est un utilisateur qui accède à l'environnement en vu de bénéficier des ressources qu'il offre sans s'engager dans une situation d'apprentissage avec un groupe d'apprenants. Par exemple, un utilisateur peut utiliser un supercalculateur partagé pour exécuter ses applications, ou bien accéder à un objet d'apprentissage donné tels que les cours.

- **Utilisateur appartenant à une communauté** : c'est la catégorie d'utilisateurs pour laquelle l'environnement est conçu, i.e. les apprenants qui veulent se former dans un domaine donné. Chaque apprenant peut faire partie d'une ou de plusieurs communautés selon son domaine d'intérêt. Pour q'un apprenant fasse partie de la communauté il doit s'inscrire à travers des outils qui sont fournis par l'environnement et qui vont être présentés par la suite.

4.3.1.3. Les ressources

Ce sont les ressources matérielles et même logicielles qui appartiennent aux différentes universités et institutions. Notre objectif est leur agrégation afin de rendre disponible un nombre important de ressources qui peuvent être partagées par différents utilisateurs. Nous donnons ci-dessous quelques exemples de ressources :

- **Les ordinateurs (PCs) :** les ordinateurs des universités et des établissements, et même les ordinateurs des différents utilisateurs de l'environnement peuvent servir comme

des ressources pour les utilisateurs pouvant être exploitées par exemple pour exécuter des tâches qui nécessitent de grandes puissances de calcul.

- **Les supercalculateurs :** chaque communauté peut disposer d'un ensemble de supercalculateurs qui peuvent être partagés et utilisés par les utilisateurs selon la nature de leurs applications.
- **Les équipements spécifiques :** tel que le télescope, microscope, bras robotique…, ces équipement peuvent être utilisés et partagés par les utilisateurs de l'environnement.
- **Les logiciels :** chaque communauté peut avoir des logiciels qui ne sont pas disponibles dans les autres communautés tel que les logiciels dont les prix de licence sont élevés. Par conséquent, nous donnons la possibilité de partager ces logiciels.
- **Les données :** les données sont hétérogènes et de différentes natures et distribuées, cependant les différents utilisateurs peuvent y accéder.

4.3.2. LA STRUCTURE DES UTILISATEURS DE L'ENVIRONNEMENT

L'unité globale de notre environnement est la communauté virtuelle et l'unité élémentaire est l'apprenant. Notre environnement doit assurer la collaboration entre les apprenants qui appartiennent à des communautés différentes, la collaboration entre les apprenants se fait en général en construisant un groupe. La figure 4.2 montre la structure des utilisateurs de l'environnement.

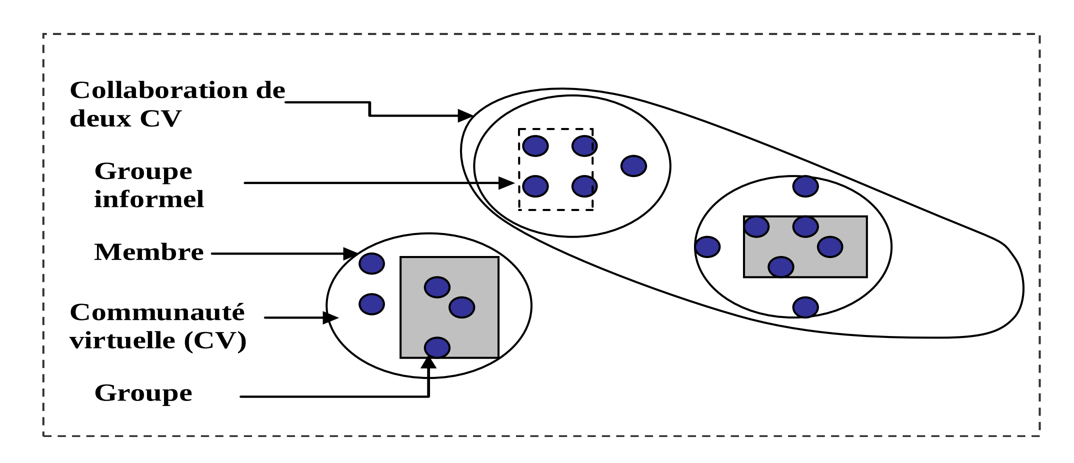


Figure 4. 2 La structure des utilisateurs

4.3.2.1. A l'intérieur d'une communauté

Dans chaque communauté l'apprentissage se fait en construisant des groupes, chaque groupe est formé d'un nombre donné d'apprenants, ces apprenants ont tous un objectif commun à atteindre qui est l'objet de la construction du groupe. Nous distinguons ici deux types de groupes : groupes formels et groupes informels **[Gro09]**.

- **Le groupe formel :** Il est constitué de membres «officiellement» désignés en vue de remplir un rôle précis, ils peuvent être permanents ou temporaires.
- **Le groupe informel :** Il se crée sans reconnaissance « officielle », par la volonté d'individus qui ont des affinités, des intérêts communs d'ordre professionnel ou personnel.

En effet, la durée de vie d'un groupe dépend de son atteinte de l'objectif commun. Le groupe se crée en définissant l'objectif à atteindre qui peut être ici la maîtrise d'un concept donné en échangeant les points de vues, les informations, en discutant, etc., donc les membres du groupe collaborent ici pour l'atteinte de l'objectif commun via les outils de collaboration. Lorsque l'objectif final est atteint, le groupe peut se dissoudre ou bien ils peuvent passer à définir un nouvel objectif et les mêmes étapes se reproduisent. Le cycle de vie d'un groupe quelconque est présenté par la figure 4.3.

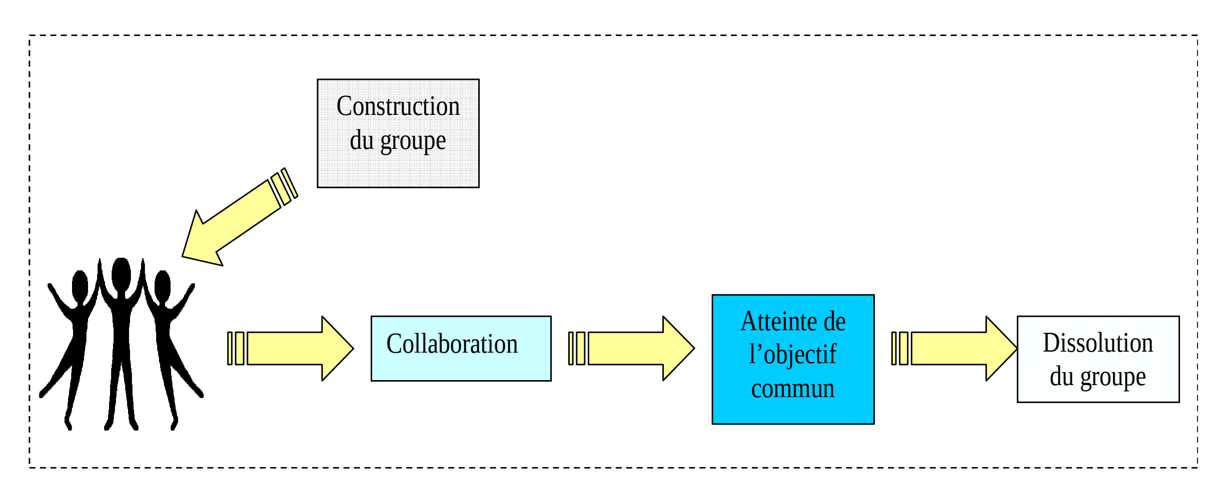


Figure 4. 3 Cycle de vie d'un groupe

4.3.2.2. A l'extérieur d'une communauté

De même qu'à l'intérieur de la communauté des groupes d'apprenants formels ou informels se forment. Les membres d'un groupe appartiennent à des communautés différentes. Notons ici qu'il est possible de faire collaborer deux ou plusieurs groupes qui appartiennent chacun à une communauté donnée.

4.3.3. LES TYPES D'INTERACTIONS EXISTANTES

Dans notre environnement de collaboration il existe plusieurs types d'interaction pouvant apparaître soit à l'intérieur d'une communauté ou entre plusieurs communautés.

4.3.3.1. Interactions au niveau de la communauté

Au niveau de la communauté nous distinguons les interactions suivantes qui sont schématisés dans la figure 4.4 ci-dessous.

- *Interaction Apprenant/Contenu* par laquelle celui qui apprend acquiert de nouvelles connaissances à partir du support qu'il utilise,
- *Interaction Apprenant/Interface* qui est la compréhension de l'utilisation de l'interface lors des transactions,
- *Interaction Apprenant/Enseignant* qui concerne le retour des résultats à l'apprenant et la communication avec l'enseignant,
- *Interaction Apprenant/Apprenant* qui permet l'échange d'informations et d'idées entre apprenants **[GNA+04]**.
- *Interaction Apprenant/Ressources ou Services.*

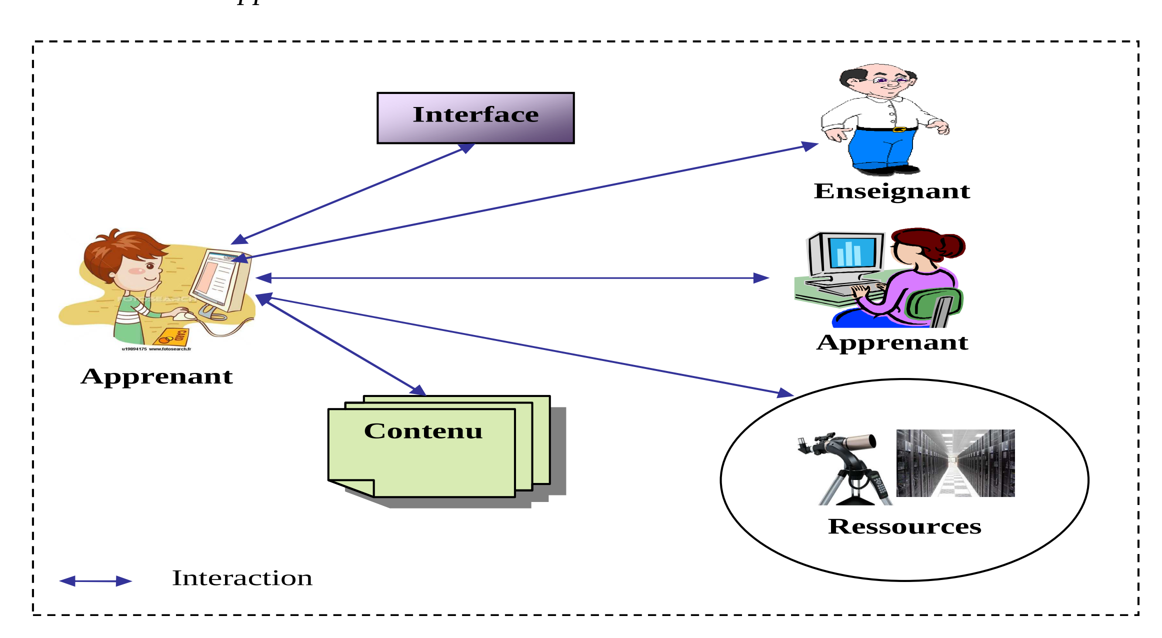


Figure 4. 4 Les interactions à l'intérieur d'une communauté

4.3.3.2. Interactions entre communautés

Les mêmes interactions qu'a l'intérieur d'une communauté peuvent se faire entre plusieurs communautés, de ce fait chaque entité appartenant à une communauté peut interagir avec n'importe quelle entité d'une autre communauté, nous citons ici quelques exemples d'interactions:

- Une interaction entre des enseignants appartenant à des communautés différentes par besoin par exemple de planifier des sessions d'évaluation commune aux classes identiques.

- Interaction d'un groupe d'une communauté donnée avec un enseignant d'une autre communauté.
- Apprenant qui peut avoir accès à des ressources (supercalculateur, logiciel…) d'une autre communauté.

4.3.4. LES NIVEAUX DE COLLABORATION

Dans notre environnement nous distinguons plusieurs niveaux de collaboration qui peuvent apparaître entre les différentes entités formant ou utilisant le système.

4.3.4.1. Collaboration entre les apprenants

Dans ce type de collaboration nous désignons l'apprentissage collaboratif entre les apprenants qui sont en réalité organisés en groupes.

4.3.4.2. Collaboration entre les groupes

Ici, la collaboration peut apparaître entre les groupes qui appartiennent à la même communauté ou bien à des communautés différentes. Parmi les raisons de ce genre de collaboration nous citons le partage du même objectif ou des objectifs similaires.

4.3.4.3. Collaboration entre les enseignants

Les enseignants qui peuvent appartenir à la même communauté ou à des communautés différentes collaborent ensemble pour échanger des informations sur la matière enseignée, pour l'élaboration d'activités pour les apprenants, ou encore l'évaluation de ces derniers, et aussi reformuler les objectifs à atteindre lorsque ces derniers ne sont pas atteints lors du processus d'apprentissage.

4.3.4.4. Collaboration entre les communautés

La collaboration entre les communautés peut apparaître via le partage des ressources de différentes natures et le partage d'enseignants. La collaboration permet aux communautés d'atteindre leur objectif final aicvnsi que les objectifs locaux propres à chaque communauté.

4.3.5. LES EXIGENCES D'UN ENVIRONNEMENT DE COLLABORATION

Dans cette partie nous allons définir les exigences d'un environnement de collaboration qui vont être modélisées dans notre environnement comme des services. Les points présentés ci-dessous sont les résultats achevés par le projet EleGI (European Learning Grid Infrastructure) et qui sont synthétisés dans **[DLJC06]**.

- **Un environnement sécurisé et fiable** : une des difficultés majeures de la collaboration est que les collaborateurs ne se connaissent pas, ce qui pose un manque de confiance entre les

membres, par exemple, est ce que la personne X est bien la personne qu'elle dit être? comment s'assurer que les messages n'ont pas été altérés ? ... Par conséquent le premier critère à assurer dans un environnement de collaboration est la sécurité et la fiabilité.

- **Collaboration synchrone et asynchrone** : ces deux modes de collaboration sont importants dans tout environnement d'apprentissage et l'un d'eux ne peut remplacer l'absence de l'autre. Le premier mode est extrait de la réalité, c.-à-d. la collaboration synchrone donne l'impression aux collaborateurs qu'ils ont entrain de collaborer face à face dans le même endroit. Le deuxième mode (collaboration asynchrone) est indispensable pour l'apprentissage à distance et il est avantageux lors de l'absence de l'apprenant. Les forums par exemple aident les apprenants à collaborer sans obliger les participants d'être présents en même temps.
- **Facilité d'accès** : l'accès à l'environnement de collaboration doit être une tâche simple sans aucune peine imposée à l'utilisateur.
- **L'ubiquité** : avoir accès à l'environnement de n'importe quel endroit et à n'importe quel moment.

En plus des points que nous venons de citer, nous voyons important d'ajouter les deux points suivants :

- **Avoir à chaque moment des informations sur tout le système** : un des points les plus importants dans un environnement collaboratif est de permettre aux participants d'être au courant de tout ce qui se passe dans le système (les personnes connectées, les personnes qui se déconnectent, les ressources disponibles, les sessions qui se déroulent…). Ce point va aider de manière indirecte les participants à collaborer et par conséquent il va faciliter la tache d'apprentissage, ce point sera détaillé ultérieurement.
- **La collaboration intra et extra organisationnelle :** notre environnement de collaboration ne doit pas être limité à la collaboration entre les membres appartenant à la même organisation ou communauté mais il doit offrir la possibilité aux membres des différentes organisations de collaborer.

4.4. ARCHITECTURE PROPOSEE

Dans la section précédente nous avons montré de manière globale les caractéristiques de notre environnement de collaboration en répondant aux questions : qui va l'utiliser ? Qu'est ce qu'il nous offre ? Que doit il assurer ? Dans cette section nous présentons son architecture.

4.4.1. VUE GLOBALE DE L'ENVIRONNEMENT

Afin de répondre au problème posé dans la problématique et après avoir expliqué notre choix de Grid comme infrastructure de base (voir chapitre 3), nous présentons ici une vue globale de notre architecture, une présentation plus détaillée sera donnée dans la section suivante.

L'objectif de cette architecture n'est pas de fournir un environnement de E-learning avec tout ce qu'il peut contenir comme services et composants, mais c'est plutôt un environnement de collaboration qui va faire réunir des systèmes de E-learning distants. L'architecture est organisée en couches, ces dernières représentent un ensemble de fonctions et chaque couche fait appel à n'importe quelle fonction des couches inférieures, la figure 4.5 montre une vue globale du système.

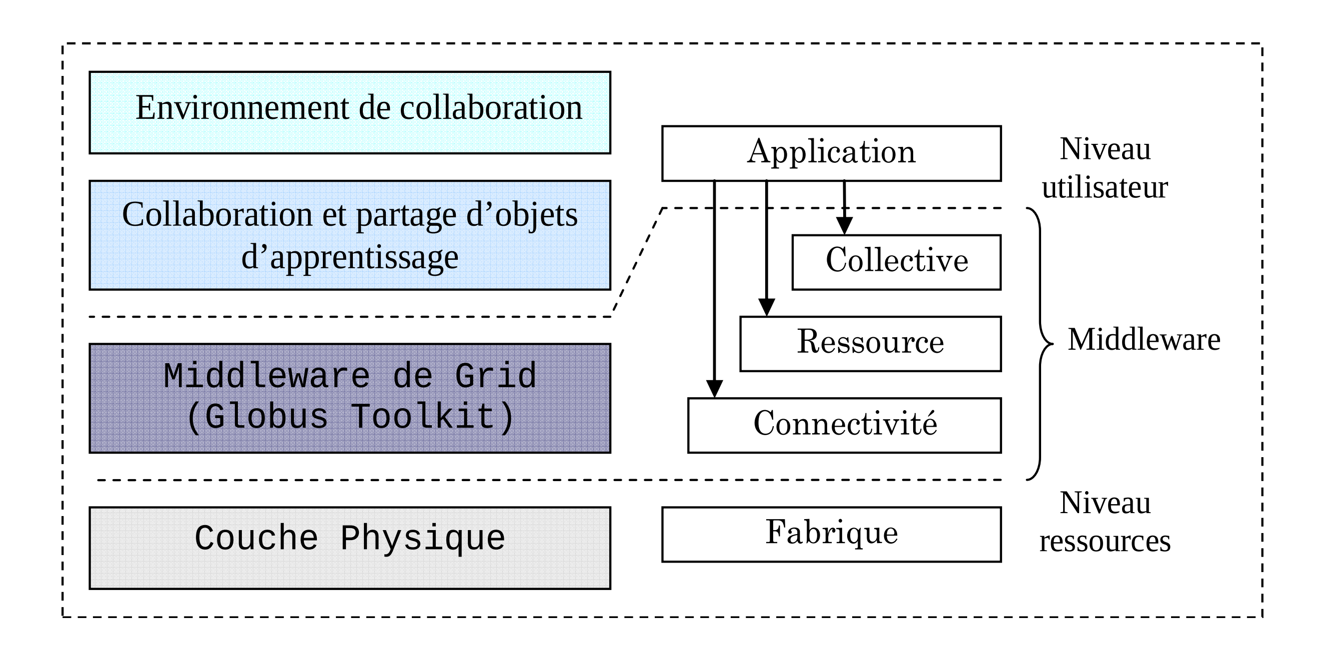


Figure 4. 5 Vue globale du système

4.4.2. DESCRIPTION DES COUCHES

Dans ce qui suit nous décrivons de manière détaillée chaque couche de notre architecture en commençant de bas en haut.

4.4.2.1. La couche physique

Contient toutes les ressources de l'environnement (ressources de calcul, ressources de Stockage, bases de données,…). Les ressources sont géographiquement distribuées et hétérogènes, cette héterogenieté est masquée par la couche middleware. La couche physique regroupe l'ensemble des ressources partagées entre les différents participants. Par exemple, les cours

disponibles dans le système ayant plusieurs formats (texte, vidéo, audio) sont stockés à ce niveau là.

Cette couche forme une agrégation de toutes les ressources qui appartiennent aux différentes communautés d'apprentissage. La figure 4.6 présente un exemple sur les types de ressources qui peuvent faire partie de notre système et la manière dont elles sont organisées.

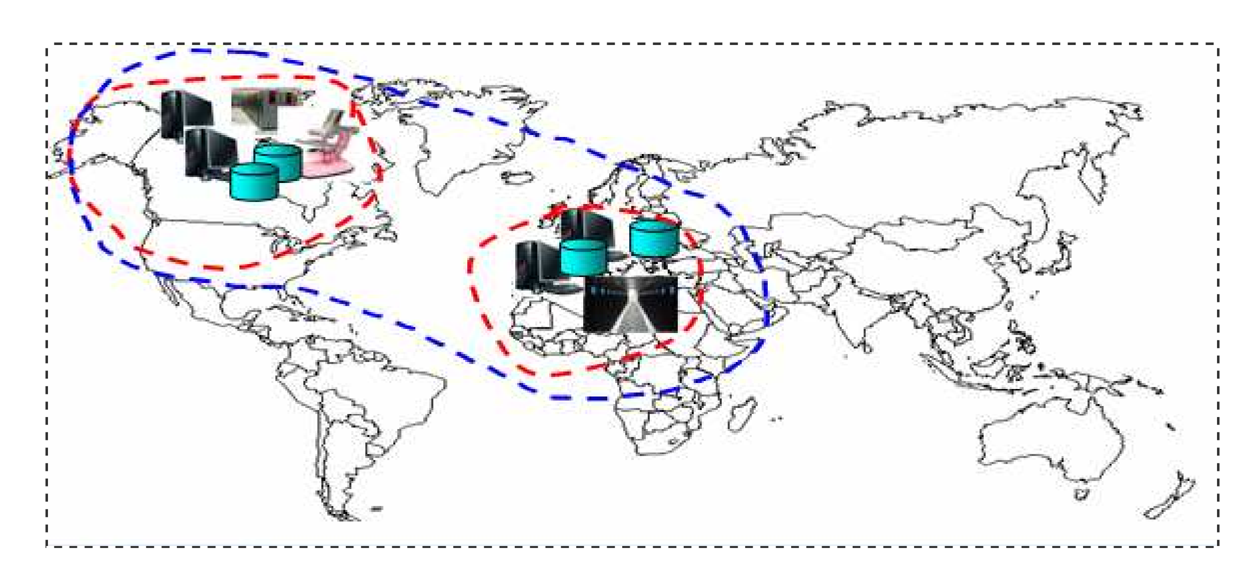


Figure 4. 6 La couche physique contient les ressources distribuées ainsi que la connexion de ces ressources

4.4.2.2. La couche Middleware

C'est l'élément de base de Grid, il permet de virtualiser l'accès aux ressources hétérogènes et distribuées, le transfert des données entre les différents nœuds de Grid et la gestion de la communication, etc. Cette couche fournit un ensemble de services qui répondent aux exigences de OGSA et qui sont implémentés par l'intergiciel Globus que nous avons choisi.

Pour bien illustrer le rôle de l'intergiciel Globus dans notre environnement nous allons présenter une partie des services qu'il offre, et comment ces derniers sont exploités pour répondre à nos besoins dans un environnement de E-learning.

Service de sécurité (GSI)

Les mécanismes de sécurité offerts par l'intergiciel Globus permettent aux administrateurs des différentes OVs (qui présentent ici des communautés virtuelles d'apprentissage (VLCs)) de gérer leurs utilisateurs et leur accès aux ressources par l'utilisation des services d'authentification et d'autorisation.

Services de gestion de données

Les services de gestion de données sont utilisés pour gérer le transfert des données entre les nœuds d'une VLC et même entre les nœuds des différentes VLCs. Comme expliqué dans le

chapitre 2, plusieurs services sont offerts pour la gestion, transfert et localisation de données. Nous donnons ci-dessous une brève présentation de ces services :

- Le **GridFTP** permet le transfert sécurisé des données entre les machines de Grid, en se basant sur l'infrastructure de sécurité. Par conséquent, tout transfert de données entre les communautés se fait via ce service.
- Le **RFT** est un web service basé sur WSRF qui utilise GridFTP pour transférer de grandes quantités de données. Il offre plusieurs avantages, tel que donner la possibilité à un client d'avoir des informations sur l'état du transfert, le rétablissement du transfert lorsque un problème survient, etc.
- Le **RLS** est un service qui maintient et donne accès aux informations sur l'emplacement physique des fichiers et données répliqués.
- Le **DRS** est un service de haut niveau qui utilise les deux services RFT et RLS pour la réplication des fichiers en assurant leur découverte.

Service d'exécution des tâches (GRAM)

Le service d'exécution des tâches GRAM offert par Globus permet aux apprenants d'exécuter leurs propres tâches dans les ressources de Grid sans savoir où et comment l'exécution se fait. En effet, nous distinguons ici deux cas différents qui font appel à ce service.

- L'exécution des tâches sur des ressources distantes peut être une tâche qui est programmée au préalable dans l'environnement, c.-à-d. que dans les objets d'apprentissage existants dans l'environnement il se peut que l'objet d'apprentissage utilisé par un apprenant ne se limite pas à un cours simple mais il possède des exercices ou bien une application qui fait appel aux ressources de Grid pour qu'elle s'exécute, tel que définit dans **[PV03]**.
- L'exécution d'une tâche donnée peut être une opération initialisée par un apprenant donné ou bien un utilisateur externe (2ème type d'utilisateur) qui veut exécuter ses applications et dont les capacités locales de sa machine ne sont pas suffisantes pour obtenir les résultats attendus. l'utilisateur sera alors responsable de faire les configurations nécessaires pour l'exécution de sa tâche en utilisant le service GRAM.

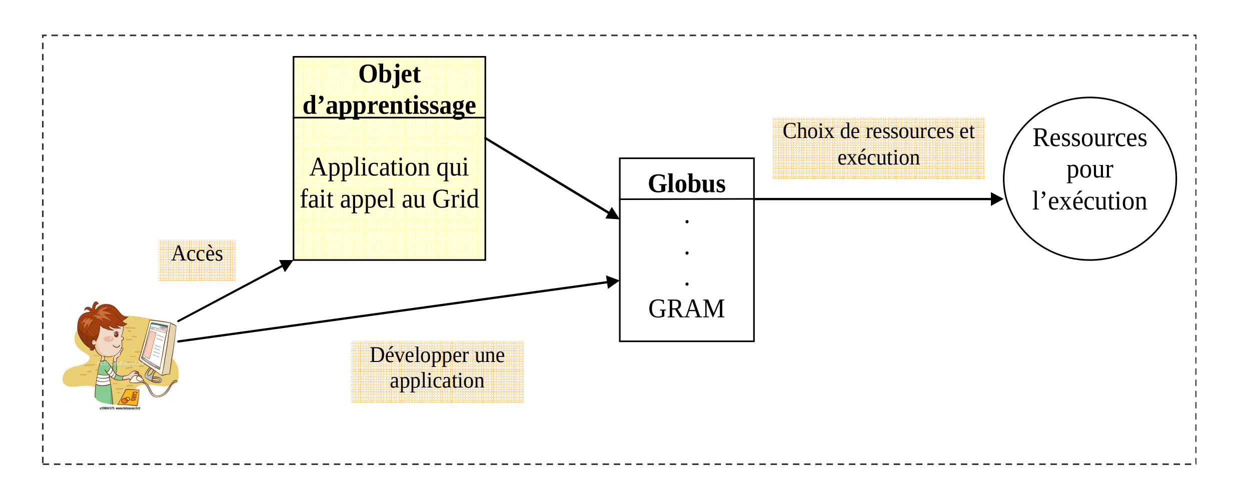


Figure 4. 7 L'exécution des tâches via le service GRAM

Service d'information (MDS)

Le service d'information est utilisé par toutes les entités de l'environnement pour obtenir une variété d'informations requises. Les informations collectées sont utilisées soit pour découvrir de nouveaux services ou ressources ou pour permettre la gestion et le contrôle de l'état du système.

- Les apprenants par exemple, peuvent l'utiliser pour savoir quels sont les services existants ? qui est connecté ? les sessions en cours d'exécution ?
- Les administrateurs de l'autre coté peuvent l'utiliser pour la gestion et le contrôle du système, il leur permet par exemple de savoir s'il y'a une surcharge réseau et de collecter des informations sur les ressources de Grid tel que l'état.

4.4.2.3. La couche collaboration et partage d'objets d'apprentissage

Cette couche s'occupe de la gestion des objets d'apprentissage existants et fournit aussi les outils de base de communication et de collaboration entre les communautés lorsque ces dernières sont connectées à l'environnement. En effet, cette couche doit présenter un middleware pour une collaboration à grande échelle à des communautés d'apprentissage qui utilisent des plateformes différentes. La collaboration est un terme très large et peut être employé dans différents niveaux. Nous mettons l'accent ici sur deux niveaux principaux ; la collaboration physique (technique) via le partage de ressources distribuées et hétérogènes, et la collaboration à un niveau plus haut (niveau méta) qui s'agit de la communication et les interactions effectuées entre les membres des communautés différentes pour la résolution en commun d'un problème donné.

Deux principaux composants sont distingués dans cette couche, *service de partage d'objets d'apprentissage* et *service de collaboration.* Cette couche est basée sur l'architecture OGSA et

utilise les services qu'elle offre qui présentent des services de bas niveau pour fournir de nouveaux services qui répondent à de nouveaux besoins.

Services de partage des objets d'apprentissage

Ce composant s'occupe de la gestion du partage des objets d'apprentissage entre différentes communautés et la coordination de ce partage.

- Permet la gestion des ressources en assurant un service d'inscription de nouvelles ressources et de suppression de ressources anciennes. Ces deux services sont nécessaires car l'environnement est caractérisé par sa nature dynamique ; à l'instant t on a un nombre donné de communautés, de ressources et d'utilisateurs et à un instant t+1 ce nombre est variable. Par conséquent, cette couche permet de gérer dynamiquement l'insertion et la suppression des ressources. Ce service fonctionne en coopération avec le service MDS qui lui donne des informations sur l'état et la structure de l'environnement.
- Contient aussi une ontologie des ressources partagées qui facilite la recherche des objets d'apprentissage par les utilisateurs de l'environnement. Il permet aussi la mise à jour de cette ontologie lorsque une nouvelle ressource est détectée ou supprimée.
- Afin de permettre la réutilisation et l'échange des contenus d'apprentissage entre les différents LMS cette couche utilise les formats SCORM (Sharable Content Object Reference Model). En effet, SCORM offre les mécanismes nécessaires pour une utilisation répétée et pour le partage de cours afin de réduire le temps et le coût pour le développement des cours, et les rendre réutilisables et acceptables pour différents LMSs **[Che08, LTT08]**.

Services de collaboration

Ce composant contient plusieurs services qui sont liés à la communication et la collaboration tels que les outils de communication et de collaboration synchrones (chat, vidéoconférence, classes virtuelles,..) ou asynchrones (e-mail, forum,…), et des outils pour la gestion des sessions de collaboration.

Comme indiqué dans le premier paragraphe de cette section, la couche collaboration et partage de ressources d'apprentissage doit présenter un middleware pour la collaboration entre plusieurs communautés qui possède chacune son propre LMS, son domaine administratif et sa politique de gestion de sécurité. L'architecture OGSA qui se situ juste au dessous de cette couche lui fournit plusieurs services tels que: l'infrastructure de sécurité GSI qui résout le problème d'héterogenieté des politiques de sécurité, le Single Sign On (SSO) pour une authentification unique pour l'accès à plusieurs ressources.

De l'autre coté, penser à concevoir des outils très puissants pour la collaboration n'est pas une tâche simple et surtout lorsqu'on pense à bâtir de nouveaux outils autres que ceux existants en exploitant les potentialités de Grid. Par conséquent, les détails de cette couche feront l'objet d'une future recherche.

4.4.2.4. La couche Environnement collaboratif

L'environnement collaboratif GCEL joue initialement le rôle d'un espace de communication de groupes, c'est le lieu de rencontre des apprenants qui veulent collaborer pour la résolution d'un problème donné, pour éliminer certains conflits dans leurs connaissances acquises, pour synthétiser, etc. Un groupe est formé d'un ensemble d'apprenants, ces derniers appartiennent à une communauté virtuelle.

GCEL constitue le lieu de rencontre des différentes entités utilisatrices du système (apprenants, enseignants et administrateurs) pour qu'ils puissent communiquer et travailler ensemble. Elle leur offre un ensemble de services qui varient de l'inscription des communautés jusqu'à la dissolution de ces dernières.

Le cœur de la couche environnement collaboratif est un LMS qui permet aux utilisateurs d'accéder à l'environnement. A la différence des LMSs propres à chaque communauté, ce dernier traite chaque LMS distant comme une ressource qui contient plusieurs services et outils qui peuvent être partagés. Chaque utilisateur qui accède à l'environnement va automatiquement accéder à la couche Middleware et par conséquent bénéficier des services de l'environnement et des services qu'offre le Grid.

GCEL offre quatre services principaux notamment le service d'inscription, service d'authentification, service Image, et le service de recherche que nous allons décrire en détail dans la section suivante.

4.4.3. ARCHITECTURE DE BASE DE GCEL

Dans cette section nous présentons la couche application qui est l'ensemble des services de haut niveau fournis par l'environnement et dont l'invocation se fait à l'aide d'une interface.

Chaque communauté peut avoir ses propres services qui vont permettre aux apprenants qui lui appartiennent de suivre le processus d'apprentissage, tels que l'accès à un cours ou à une ressource donnée de la communauté selon le niveau de l'apprenant et ses droits d'accès. Chaque communauté est alors considérée comme étant un ensemble de services dédiés pour un ou plusieurs types de formation. La communauté est un fournisseur de services lorsque elle fournit des services à ses apprenants ou à d'autres communautés, la communauté est un consommateur de

services lorsque elle utilise des ressources et des services autres que les siennes pour former ses apprenants.

La figure ci-dessous montre l'architecture de base de GCEL

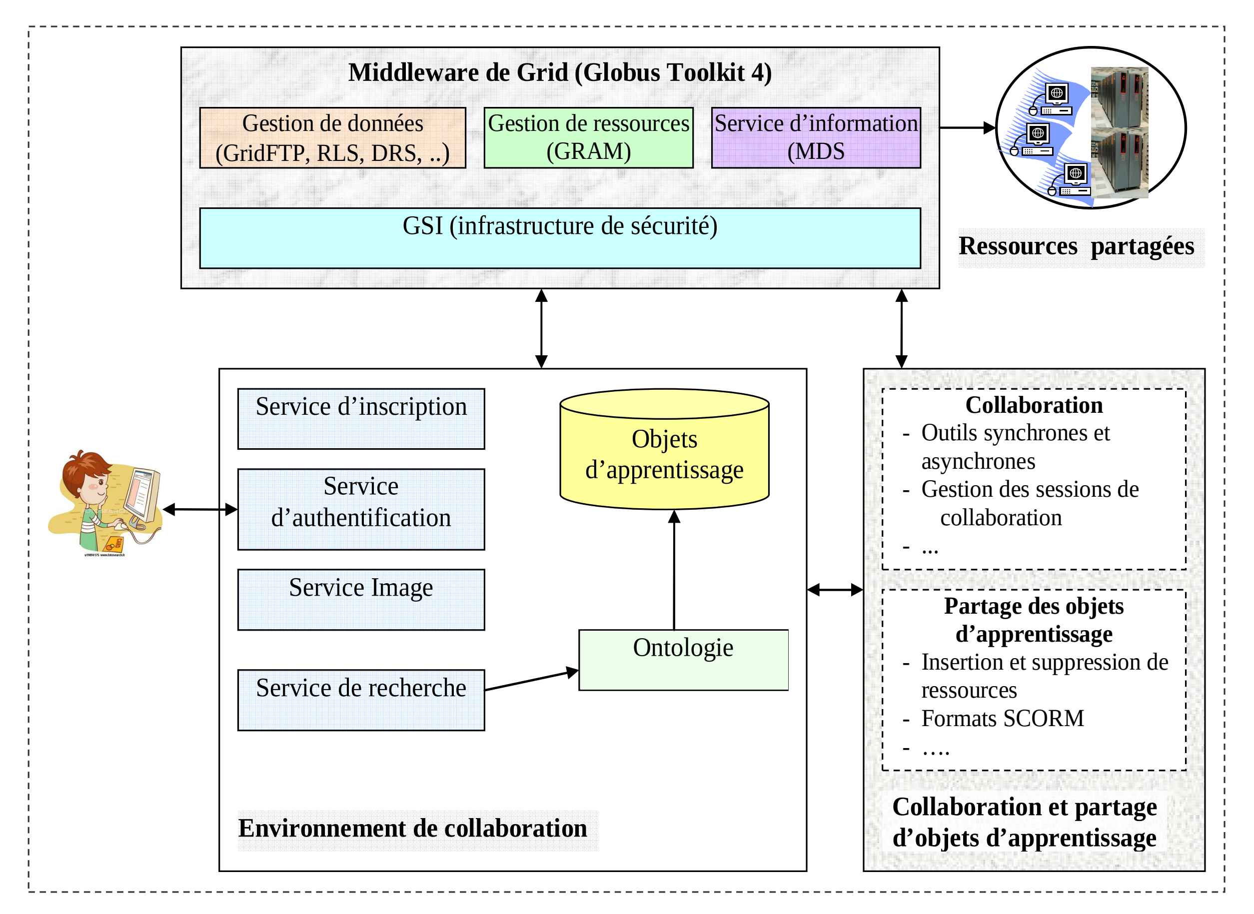


Figure 4. 8 Architecture de base de GCEL

4.4.3.1. Service d'inscription

Le service d'inscription comme son nom l'indique, donne la possibilité aux communautés de s'inscrire dans l'environnement. L'inscription d'une communauté veut dire la mise à disposition des ressources qu'elle détient aux ressources de l'environnement, et aussi bénéficier des services fournis par l'environnement tel que le partage de ressources et la collaboration avec d'autres communautés.

Lorsque une communauté est inscrite les services de la sous couche collaboration interviennent pour ajouter ses ressources aux ressources de Grid et mettre à jour les informations globale de Grid.

Ce service ne permet pas uniquement l'inscription des communautés mais aussi l'inscription des utilisateurs libres (comme expliqué dans le début de ce chapitre) qui veulent exécuter leurs applications en utilisant les ressources de Grid ou pour accéder aux contenus d'apprentissage

disponibles dans notre environnement.

4.4.3.2. Service d'authentification

Chaque utilisateur voulant accéder à l'environnement doit être authentifié, le service d'authentification utilise les mécanismes de sécurité offerts par l'infrastructure de sécurité GSI pour l'authentification des utilisateurs.

4.4.3.3. Service de recherche

Ce service permet la recherche des objets d'apprentissage à travers une ontologie des ressources partagées, l'ontologie appartient à la sous couche collaboration. En effet, pour faciliter la compréhension et le partage de connaissances entre différentes communautés d'apprentissage, nous aurons besoin d'un outil qui permet aux acteurs d'être d'accord sur les termes qu'ils utilisent. L'ontologie est l'outil choisi ici, qui va faciliter aux apprenants l'accès aux concepts qu'ils cherchent.

Nous voyons ici la nécessité de définir l'ontologie qui est à la base du service de recherche que nous venons de présenter.

Les ontologies informatiques sont des outils qui permettent précisément de représenter un corpus de connaissances sous une forme utilisable par un ordinateur **[Ont09]**. Une des définitions de l'ontologie qui fait autorité est celle de Gruber **[Gru93]**:

Une ontologie est la spécification d'une conceptualisation d'un domaine de connaissance.

Cette définition s'appuie sur deux dimensions :

- Une ontologie est la conceptualisation d'un domaine, c.-à-d. un choix quant à la manière de décrire un domaine.
- C'est par ailleurs la spécification de cette conceptualisation, c.-à-d. sa description formelle.

C'est une base de formalisation des connaissances. Elle se situe à un certain niveau d'abstraction et dans un contexte particulier. C'est aussi une représentation d'une conceptualisation partagée et consensuelle, dans un domaine particulier et vers un objectif commun. Elle classifie en catégories les relations entre les concepts.

4.4.3.4. Service Image

C'est un service d'information et constitue le point clé de notre environnement de collaboration, il set à enrichir le système par une variété d'informations qui sont utiles pour le bon

fonctionnement de l'environnement, il permet aussi de favoriser la collaboration entre les différents acteurs de l'environnement.

Motivations

Afin de permettre la collaboration entre les utilisateurs d'un environnement donnée la disponibilité des outils adéquats n'est pas toujours suffisante, cependant il est très important de tenir compte d'un deuxième point qui s'agit de rendre le collaboration elle-même une tache facile pour les utilisateurs. En réalité, dans la plupart des systèmes de E-learning existants, malgré la disponibilité d'une variété d'outils de collaboration synchrones (chat, vidéo conférences, ..) et asynchrones (forum, mail...) le problème de collaboration persiste encore.

Si nous supposons un ensemble d'apprenants appartenant aux différentes communautés et qui veulent construire un groupe pour apprendre ensemble, le problème posé est comment permettre à ces membres de choisir les tâches par lesquelles ils intéressent, soit au niveau local à la communauté ou au niveau externe entre deux ou plusieurs communautés ? Comment permettre à un apprenant donné de s'intégrer facilement aux activités d'apprentissage ? Comment informer les administrateurs des différentes communautés de l'état du réseau ?

Dans le 3ème chapitre de ce mémoire nous avons fait une petite analyse sur les services fournis par un système de E-learning, et nous avons posé une question sur la possibilité de fournir des informations en temps réel sur toutes les activités et les changements subis par le système sans aucune planification au préalable? La réponse à cette question est le service Image.

Description

Le service Image est un service qui ressemble au service d'information fourni par Globus, son rôle est d'enrichir l'environnement de collaboration par une variété d'informations en temps réel sur toutes les activités qui peuvent avoir lieu dans les différentes communautés ; les ressources disponibles, l'état du réseau, les sessions en cours de déroulement, etc.

En effet, le service image va jouer le rôle d'intermédiaire entre les différentes communautés qui utilisent des LMSs différents. Pour faciliter sa mise en œuvre nous avons distingué deux déploiement différents : *Service Image interne* et *service Image externe* ; nous utilisons les termes interne et externe par rapport à la communauté. Le service image externe appartient à l'environnement GCEL et les services Image internes jouent le même rôle que le service interne mais sont spécifiques à chaque communauté. Les services Image internes sont des sources d'information pour le service image externe, ils collectent des informations en temps réel sur ce qui se passe dans la communauté auquelle ils appartiennent. Une vision globale de ce service est schématisée sur la figure 4.9.

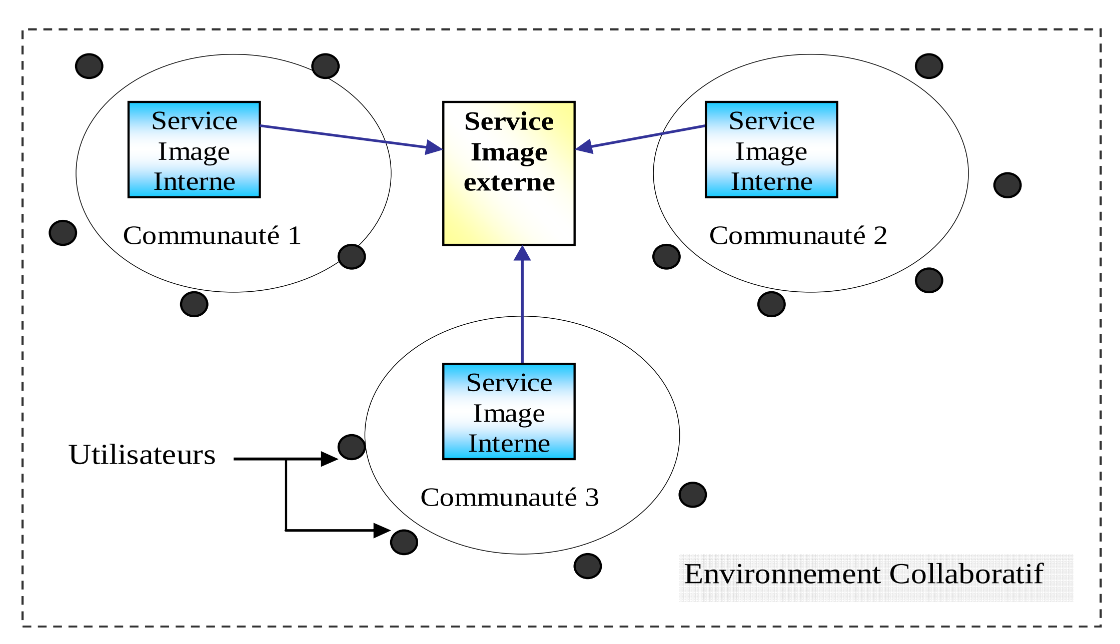


Figure 4. 9 le Service Image favorise la collaboration dans une communauté et entre plusieurs communautés

- **Service Image Interne**

Contient toutes les informations concernant les activités s'effectuant dans le système de E-learning, par exemple classe virtuelle, vidéo conférence, groupes disponibles, etc. l'idée de base de ce service est de donner à tout moment (période fixe) une image ou une vue sur tout le système afin de permettre aux apprenants de joindre facilement leurs points d'intérêt, trouver sans peine les collaborateurs et savoir toute tâche qui se déroule au niveau du système. Ce service est mis à jour dynamiquement.

- **Service Image Externe**

Ce service sert à favoriser et faciliter la collaboration entre plusieurs communautés, il sert d'intermédiaire entre plusieurs services Image internes, il se situe au niveau méta. Ce service peut être utilisé dans deux contextes différents:

  - Favoriser la collaboration entre apprenants qui appartiennent aux communautés différentes.
  - Faciliter la collaboration entre les administrateurs des différentes communautés, les concepteurs de cours et les enseignants.

Le service Image est basé sur le Grid en bénéficiant des services qu'il offre. Il utilise le service d'information MDS (Meta Directory Service) fourni par Globus qui est considéré comme une source d'information. Il utilise aussi le GridFTP pour le transfert sécurisé des données entre les différentes communautés.

Les niveaux d'information

Dans le service Image nous distinguons plusieurs niveaux d'informations, et chaque catégorie d'utilisateur (administrateur, apprenant, enseignant) peut accéder à un ou plusieurs niveaux d'informations selon ses droits.

- Evénements globaux sur le fonctionnement et l'état physique de la plateforme (ressources disponibles, ressources consommées, surcharge réseau, les nœuds déconnectés, …).
- Evènements sur les services disponibles et qui sont en cours d'exécution (chat, vidéo conférences, transfert de fichiers, …).
- Evénements sur les différents membres de la communauté (apprenants, enseignants, administrateurs, …).
- Evénements sur les activités qui se déroulent entre les communautés (réunions, évaluation, cours en ligne ou classe virtuelle,…).

Fonctionnement du service Image

L'utilisateur faisant partie du système qui peut être : apprenant, enseignant, ou bien administrateur interagit avec le service Image à l'aide d'une interface graphique, puis choisit le type ou bien le niveau d'information désiré, sachant que les informations fournies dépendent du type d'utilisateur ; un apprenant par exemple ne peut pas consulter les informations qui concernent l'état et la structure de Grid, en revanche c'est l'administrateur qui en a le droit. Finalement, le service va répondre par une série d'informations en temps réel sur l'information requise. Après avoir obtenu l'information, l'utilisateur va réagir en s'intégrant par exemple à une session d'apprentissage qui se déroule sur un sujet donné.

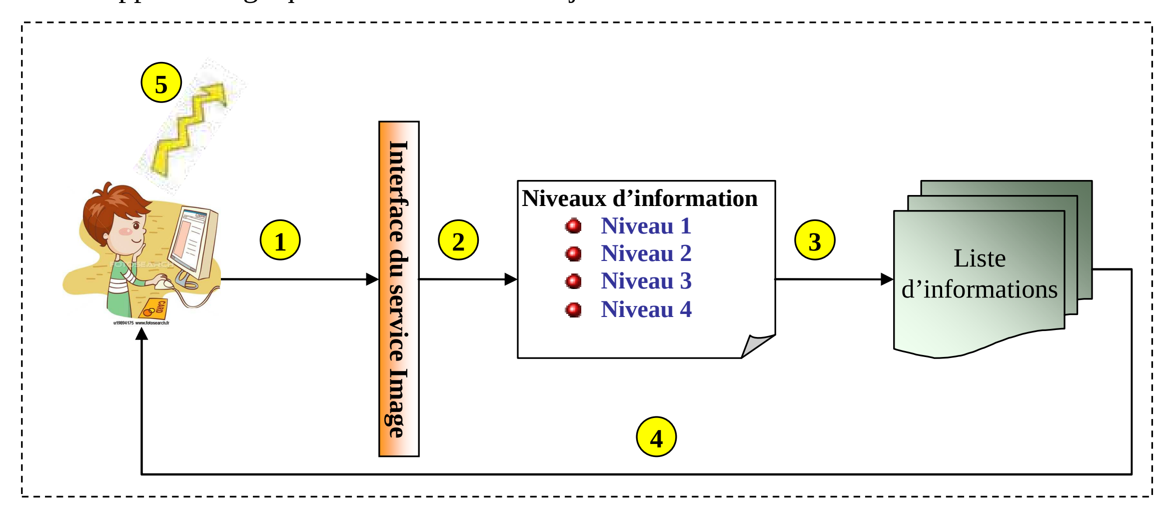


Figure 4. 10 Fonctionnement du service Image

4.5. PRINCIPE DE FONCTIONNEMENT DE GCEL

Nous supposons un ensemble de communautés de formation où chacune a ses propres ressources (humaines, matérielles et logicielles) et son propre LMS qui permet de gérer les différentes activités qui se déroulent au niveau de la communauté, tels que la gestion des utilisateurs et la gestion d'accès aux ressources. Afin de permettre à ces communautés de travailler ensemble sur l'environnement GCEL, un ensemble d'étapes doit être suivi qui se résume ainsi :

- La communauté doit d'abord spécifier les ressources à partager avec les autres communautés : une communauté donnée dispose d'un ensemble de ressources qui sont utilisées durant le processus d'apprentissage, pour qu'elle s'allie à une communauté donnée elle doit spécifier les ressources à partager parmi l'ensemble de ses ressources, comme elle peut partager toutes ses ressources, cela se fait à travers des négociations avec les communautés concernées.
- Chaque communauté utilise son propre LMS, cependant chaque plateforme LMS utilise son matériel qui ne peut pas être échangé avec les autres LMSs d'où la nécessité de faire appel à d'autres standards afin de résoudre le problème d'héterogenieté du contenu et de les rendre réutilisables tels que les formats SCORM. Les contenus d'apprentissage seront alors stockés dans notre plateforme selon un niveau de granularité qui permet leur réutilisation et échange.
- Un apprenant d'une communauté donnée lorsqu'il veut collaborer avec un apprenant d'une autre communauté, doit s'authentifier d'abord dans l'environnement de collaboration pour avoir la possibilité de continuer son processus de collaboration ou d'accès aux ressources.
- Si nous supposons que chaque communauté a son propre LMS, dans ce cas chaque communauté a sa pédagogie suivie durant le processus de formation, ses politiques de sécurité et son domaine administratif, par conséquent toute collaboration qui se fait entre les membres de la même communauté se fait en utilisant les services offerts par l'LMS utilisé en plus des services (authentification, transfert de données…) offerts par Globus qui sera obligatoirement déployé dans chaque site ,et dans ce cas chaque communauté constitue un nœud de Grid. La communication entre les membres d'une communauté se fait par les services de collaboration synchrones ou asynchrones offerts par la communauté.

- Cependant, dès qu'il y a une communication entre les membres appartenant à des communautés différentes cela ne se fait pas à travers les services de chaque communauté mais a travers les services de notre environnement de collaboration dont l'appel à chaque service signifie son instanciation.

4.6. QUELQUES SCENARIOS

Dans cette section nous présentons quelques scénarios qui permettent de clarifier le principe de fonctionnement de l'environnement GCEL

4.6.1. L'ACCES A UN CONTENU DISTANT

La figure 4.11 ci-après illustre le principe subi pour l'accès au contenu.



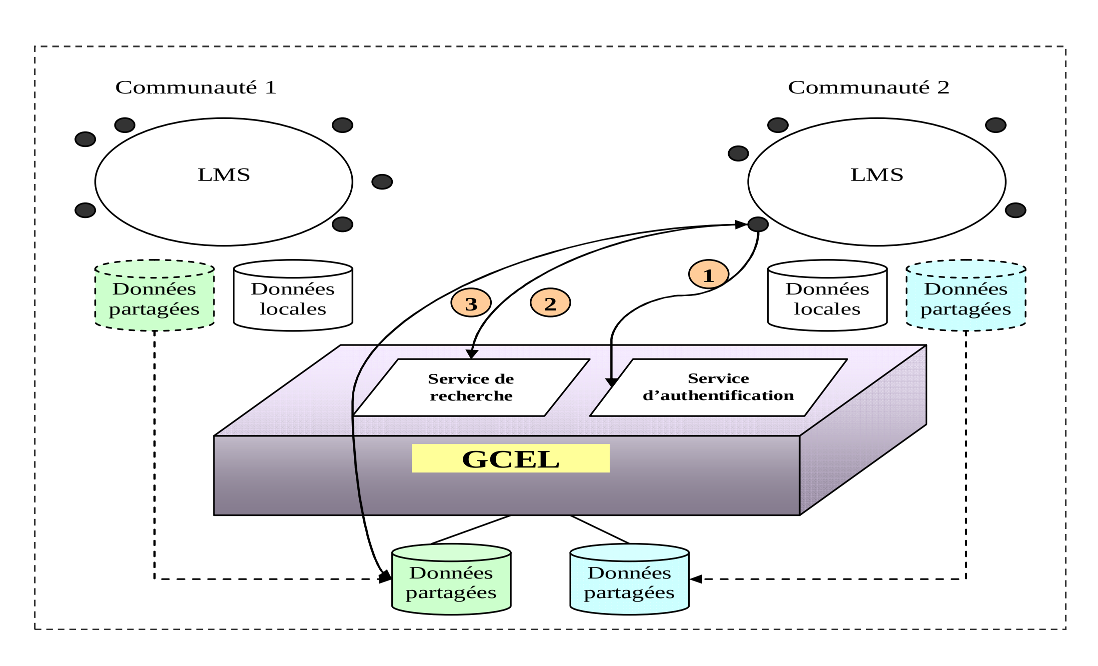


Figure 4. 11 La recherche et l'accès à un contenu appartenant aux ressources de Grid

Dans une des sections précédentes de ce chapitre, nous avons mentionné que la communauté peut partager tout ou partie de ses ressources. Par conséquent, si un apprenant donné accède à un contenu privé à sa communauté cela se fait sans aucune authentification et c'est le LMS de la communauté qui va se charger de la gestion de l'accès au contenu.

Lorsque une communauté est connectée, ses membres peuvent bénéficier de tous les services que l'environnement offre, mais l'accès à un contenu donné de la plateforme nécessite d'abord une authentification et selon ses droits d'accès, l'utilisateur peut accéder au contenu d'apprentissage désiré.

4.6.2. ROLE DES SERVICES DE GLOBUS POUR L'ACCES AUX RESSOURCES DE L'ENVIRONNEMENT

Pour l'accès à une ressource de Grid soit pour exécuter une tache ou pour accéder aux données, La première étape consiste en une authentification de l'utilisateur via le *service d'authentification* en comparant le nom et le mot de passe de l'utilisateur avec les informations stockées dans la base de données et ceci à travers le GSI. Lorsque l'utilisateur est authentifié dans ce cas le système va lui montrer une liste de toutes les ressources disponibles, et L'utilisateur dans ce cas peut lancer sa requête. Le service MDS retourne des informations sur les ressources disponibles et leur état, le GridFTP permet le transfert des données requises par l'utilisateur de manière sécurisée, et pour un accès rapide aux ressources, le RLS supporte différents emplacement pour le même fichier dans le Grid.

Nous remarquons ici que les services offerts par Globus interviennent dans chaque opération qui peut avoir lieu dans l'environnement. La figure ci-dessous montre de manière plus précise le rôle de la couche middleware (GT4) pour permettre à l'utilisateur d'accéder à l'objet d'apprentissage requis.

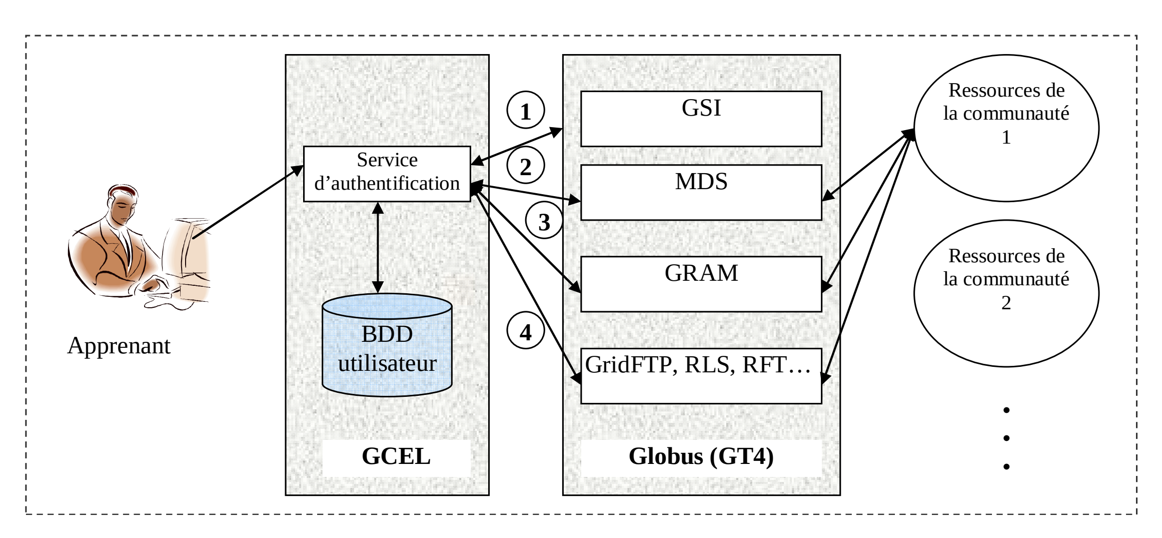


Figure 4. 12 Rôle des modules de Globus dans l'environnement GCEL

4.6.3. LA COLLABORATION ENTRE UN GROUPE D'APPRENANTS DE COMMUNAUTES DIFFERENTES

Si deux ou plusieurs membres veulent communiquer ou bien collaborer, ils doivent d'abord s'authentifier dans l'environnement de collaboration à travers le service d'authentification, la couche collaboration et partage de ressource est responsable de la gestion de la session de collaboration initiée par le groupe, dès qu'une session est initiée les services formant l'environnement sont instanciés dans le conteneur de services et peuvent être utilisés.

4.7. OUTILS POUR LA MISE EN ŒUVRE

Pour la mise en œuvre de notre plateforme l'intergiciel Globus Toolkit 4 doit être installé sur chaque site en suivant le processus d'installation qui est décrit en détail dans l'annexe. Ensuite, nous installons dans chaque site les différents LMS existants pour stocker et gérer le contenu d'apprentissage de chaque site comme Claroline et Moodle. Dans un serveur donné l'environnement GCEL sera installé et l'accès des communautés à l'environnement GCEL se fait à travers une interface Homme/machine fournie par l'environnement de collaboration GCEL et chaque LMS est lié avec GCEL via une interface.

Il est à noter que le LMS tel qu'il est n'est pas suffisant mais des fonctionnalités doivent être ajoutées tel que par exemple le service Image Interne qui sera obligatoirement déployé dans chaque nœud où l'LMS est installé. La figure 4.13 montre les outils et les interconnexions nécessaires pour le fonctionnement de GCEL.

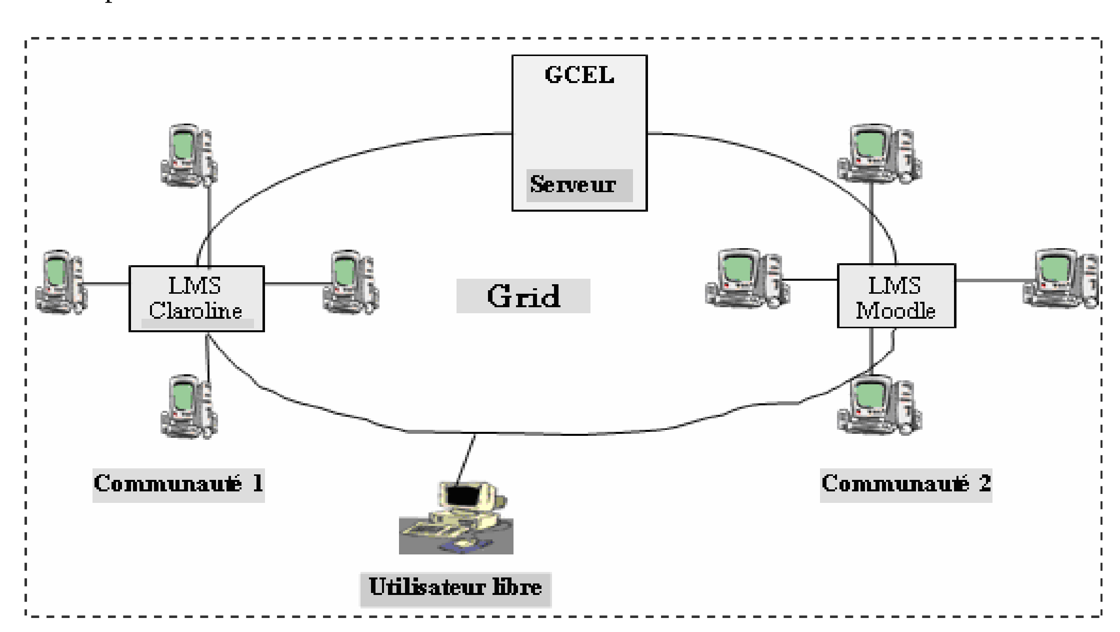


Figure 4. 13 Interconnexion de GCEL avec différents LMSs

4.8. CONCLUSION

Dans ce chapitre, nous avons présenté l'architecture de GCEL qui est un environnement de collaboration pour les communautés distantes utilisant des LMSs différents. La collaboration peut apparaître via le partage de ressources distribuées, l'apprentissage collaboratif entre un groupe d'apprenants, le partage des enseignants, etc. Cet environnement est basé sur la technologie de Grid à cause des potentialités que cette dernière peut offrir pour le E-learning et pour la collaboration.

L'utilisation de Grid a enrichi l'environnement GCEL par un ensemble très important de services et fonctionnalités tels que la sécurité (authentification via le GSI, le mécanisme SSO et la gestion des autorisations), la gestion des communautés et la réplication pour un accès rapide aux fichiers.

Pour gérer les sessions de collaboration qui peuvent être initiées à partir de GCEL, et afin de permettre le partage de contenus de différents formats, une couche de standards et de services de base a été ajoutée au dessus de la couche des services de OGSA. De ce fait, la couche de collaboration et de partage de ressources d'apprentissage bénéficie des Grid services de l'intergiciel et les considèrent comme des services de base pour son fonctionnement.

Dans le but de rendre possible l'accès à l'environnement et son utilisation, des services de haut niveau sont offerts par la dernière couche, nous citons : le service d'authentification, service d'inscription, service de recherche et le service Image. Le service Image est considéré comme un outil qui permet de favoriser la collaboration et la participation des utilisateurs et enrichi l'environnement par des informations diverses et de différentes natures.

A partir de cette proposition, nous constatons l'intérêt d'utiliser le Grid comme moyen pour arriver à une collaboration à grande échelle. En effet, deux grands avantages sont offerts par le Grid, d'un coté sa puissance de gestion des organisations virtuelles en résolvant tous les problèmes d'héterogenieté et de distribution qui sont considérés comme le premier obstacle de la collaboration, de l'autre coté l'utilisation de Grid signifie l'assurance d'un large éventail de ressources (puissance de calcul, espace de stockage et même des ressources spécifiques) aux communautés différentes, ce qui peut donner la chance aux communautés de s'améliorer.

CONCLUSION ET PERSPECTIVES

Dans le but de répondre au problème posé au début de ce manuscrit ; qui s'agit de présenter une architecture d'un environnement de collaboration qui peut être utilisé par des communautés virtuelles d'apprentissage pour des fins éducatives, nous nous sommes proposés d'utiliser le Grid comme infrastructure de base vu les avantages qu'il peut apporter soit pour le E-learning ou pour la collaboration.

En réalité, le point de départ de notre recherche était de comprendre le principe de Grid et maîtriser ses concepts et sa mise en œuvre via les intergiciels existants, et le E-learning n'est donc qu'un domaine d'application de Grid parmi plusieurs. Cependant, le choix du E-learning et plus particulièrement la collaboration dans le cadre du E-learning n'est pas fait au hasard, mais c'est à partir des informations et conclusions que nous avons pu tirées de l'état de l'art. Par conséquent, nous nous sommes trouvés devant deux points essentiels : d'un coté appliquer le Grid dans le domaine du E-learning pour monter son intérêt et son apport pour ce dernier, de l'autre coté fournir un environnement de collaboration pour le E-learning en justifiant le choix de Grid.

La recherche que nous avons faite nous a mené à réaliser et à arriver à plusieurs choses. D'abord, nous avons pu maîtriser en premier lieu les concepts liés au Grid; ses standards, domaines d'application et ses implémentations. Après, nous avons descendu à niveau plus bas qui est celui des intergiciels et plus précisément l'intergiciel Globus, que nous avons installé et testé avec succès et par conséquent un manuel d'installation a été inséré à la fin de ce mémoire qui peut servir comme un guide pour la mise en œuvre d'un nœud de Grid.

Par la suite, nous avons fait un tour d'horizon sur les concepts de E-learning et de collaboration suivi d'une synthèse de quelques travaux qui sont en effet des solutions de E-learning à base de Grid. La synthèse de ces travaux nous a montré les différentes modalités d'usage de Grid dans le E-learning et la collaboration. Enfin, cette partie a été accompagnée par une comparaison qui nous a aidé pour développer notre propre environnement.

Dans notre proposition nous avons fait la combinaison de deux domaines qui semblent indépendants : le premier s'agit de la technologie de Grid qui concerne le partage cordonné de ressources distribuées et la résolution de problèmes au sein d'organisations virtuelles dynamiques

et mutli-institutionnelles. Le second, concerne la collaboration dans le domaine de E-learning. Cette combinaison est rendue possible via l'architecture GCEL que nous avons proposée et décrite dans la partie contribution. C'est une architecture orientée service et fournit un environnement de collaboration pour des communautés virtuelles d'apprentissage utilisant des LMSs différents.

L'environnement proposé offre plusieurs avantages :

- Le partage des ressources distribuées et hétérogènes entre les communautés d'apprentissages.
- Donner aux universités ou aux institutions qui ont des budgets insuffisants la chance de s'améliorer.
- Meilleure utilisation des ressources : Les ressources de différentes natures existent mais elles ne sont pas pleinement exploitées, malgré les besoins croissants de ces puissances.
- Intégration des ordinateurs des écoles et institutions non reliés pour remplacer les serveurs.
- Utiliser les plateformes d'apprentissage (LMS) existantes et qui offrent la plupart des services nécessaires, au lieu de concevoir toute une nouvelle solution de E-learning.
- Donner aux communautés le choix d'utiliser la plateforme qui leur convient et plus particulièrement les plateformes en open source.

Tout au long de ce travail, nous n'avons pas utilisé le terme "Learning Grid" (ou "grille d'apprentissage" en français) malgré que ce terme reflète réellement ce que nous venons de traiter dans ce mémoire, car pour l'utiliser il faut avoir d'abord utilisé le Grid sémantique - la combinaison de Grid de calcul et le web sémantique - qui n'a pas été abordé ici. Malgré que Le Grid propose des améliorations significatives à nos capacités de calcul, traitement de l'information et la collaboration, et a des ambitions passionnantes dans de nombreux domaines d'activité, nous affirmons que toute la richesse de la vision de Grid, avec ses différentes modalités d'usage exige le Grid sémantique.

Parmi les perspectives envisageables à partir du système proposé, nous citons:

- Détailler la couche "collaboration et partage de ressources d'apprentissage"qui joue le rôle d'un middleware pour la collaboration à grande échelle.
- Utiliser le Grid sémantique qui a montré à travers ces dernières années son principal rôle dans le domaine de E-learning ; un nombre important des projets actuels qui utilisent le Grid sémantique sont résumés dans **[CGM+08]**.

- Essayer de passer au coté expérimentation en implémentant d'abord quelques modules. En effet, notre point de départ sera de tester l'accès aux ressources partagées depuis des nœuds de Grid différents et des LMS différents.

En résumé, le Grid dispose des capacités et fonctionnalités nécessaires pour développer des environnements de collaboration à grande échelle. Deux principaux points font des Grids un moyen pareil : leurs capacité à gérer et à maintenir les organisations virtuelles, et l'assurance d'un large éventail de ressources de différentes natures. Cependant, le Grid Computing employé seul ne résout pas entièrement le problème de collaboration, mais il doit être combiné avec d'autres solutions, tel que le Grid sémantique. Malgré que la problématique posée dans ce mémoire est assez vaste, nous avons quand même essayé de donner une solution, et de répondre aux questions posées au départ.

Enfin, une partie de notre travail a fait l'objet de deux communications internationales **[ASI09**, **AST09]**.

Bibliographie

[ABK+05]: W. Allcock, J. Bresnahan, R. Kettimuthu, M. Link, C. Dumitrescu, I. Raicu & I. Foster, "**The Globus Striped GridFTP Framework and Server**",SC'05, ACM Press, 2005.

[ACG05]: S. Aubert, P.Crescenzo & M. Gautero, "**Modélisation UML du E-learning adaptatif**", CEMAFORAD-2 Université de Bejaia: 12, 13 et 14 novembre 2005.

[AFM04]: W. E. Allcock, I. Foster & R. Madduri, "**Reliable Data Transport: A Critical Service for the Grid**", Building Service Based Grids Workshop, Global Grid Forum 11, Juin 2004.

[ASI09] : www.journal.esrgroups.org/jes/iceedt/JES&JASEselection.pdf, janvier 2010.

[AST09] : www.ecai.org/activities/**2009_**Taiwan/PNC**2009**.pdf, décembre 2009

[BCC+04]: D. Box, E. Christensen, F. Curbera, D. Ferguson, J. Frey, M. Hadley, C. Kaler, D. Langworthy, F. Leymann, B. Lovering, S. Lucco, S. Millet, N. Mukhi, M. Nottingham, D. Orchard, J. Shewchuk, E. Sindambiwe, T. Storey, S. Weerawarana & S. Winkler, "**Web Services Addressing (WS-Addressing)**", 10 August 2004. http://www.w3.org/Submission/ws-addressing/

[BCDS04]: H. Bal, H. Casanova, J. Dongarra & S. Matsuoka, "**Application-level Tools**", chapter 24 of The Grid 2: Blueprint for a New Computing Infrastructure. Morgan Kaufmann Publishers, 2004.

[BDVT04]: V. Baudin, K. Drira, T. Villemur & S. Tazi, "**A model-driven approach for synchronous dynamic collaborative E-learning**", E-Education applications: human factors and innovative approaches, Ed. C.Ghaoui, Information Science Publishing, ISBN 1-59140-292-1, 2004, Chap. III, pp.44-65.

[Bor05]: B. Sotomayor, "**The Globus Toolkit 4 Programmer's Tutorial**", 2005. http://www.chinagrid.net/grid/paperppt/progtutorial.pdf

[BTNK07]: B. Batchakui, C. Tangha, R. Nkambou & G. Kouamou, "**Environnement de collaboration basé sur le Grid Learning Services (GLS) pour les communautés de formation à centre d'intérêt commun**". In Proceedings of the 4th LORNET Annual Scientific Conference (I2LOR), Montreal, 7 November 2007.

[Cda09]: **Communauté d'apprentissage**, http://fr.wikipedia.org/wiki/Communaut%C3%A9_d%27apprentissage, janvier 2009.

[CDF+02]: A. Chervenak, E. Deelman, I. Foster, L. Guy, W. Hoschek, A. Iamnitchi, C.Kesselman, P. Kunszt, M. Ripeanu, B. Schwartzkopf, H. Stockinger, K. Stockinger &

B. Tierney, "**Giggle: A Framework for Constructing Scalable Replica Location Services**". Proceedings of the SC2002 Conference, Baltimore, November 2002.

[Cdi09]: **Communauté d'intérêt**, http://fr.wikipedia.org/wiki/Communaut%C3%A9_d%27int%C3%A9r%C3%AAt, Janvier 2009.

[Cdp09]: **Communautés de pratique et communautés d'apprentissage virtuelles : avantages, obstacles et facteurs de réussite,** http://www.elearningpapers.eu/index.php?page=doc&doc_id=10219&doclng=8, Janvier 2009.

[CFF+04a]: K. Czajkowski, D. F Ferguson, I. Foster, J. Frey, S. Graham, T. Maguire, D. Snelling & S. Tuecke, "**From Open Grid Services Infrastructure to WS-Resource Framework: Refactoring & Evolution**", March 2004, http://www-106.ibm.com/ developerworks/library/ws-resource/ogsi_to_wsrf_1.0.pdf.

[CFF+04b]: K. Czajkowski, D. F Ferguson, I. Foster, J. Frey, S. Graham, I. Sedukhin, D. Snelling, S. Tuecke & W. Vambenepe, "**The WS-Resource Framework**", Mars 2004, http://www-106.ibm.com/developerworks/library/ws-resource/ws-wsrf.pdf.

[CGM+08]: N. Capuano, M. Gaeta, S. Miranda, F. Orciuoli & P. Ritrovato, "**The Grid for learning**“, M.D. Lytras et al. (Eds.): WSKS 2008, CCIS 19, pp. 57–66, 2008.

[Che08]: T. Chen, "**Methodology for Supporting Novel Model of E-learning Platform in Grid Architecture**", F. Li et al. (Eds.): ICWL 2008, LNCS 5145, pp. 314 – 321, 2008. © Springer-Verlag Berlin Heidelberg, 2008.

[Con09]: **Condor**. http://www.cs.wisc.edu/condor/. Mars 2008.

[CPB+04]: A.L. Chervenak, N. Palavalli, S. Bharathi, C. Kesselman & R. Schwartzkopf, "**Performance and Scalability of a Replica Location Service**“. Proceedings of the International IEEE Symposium on High Performance Distributed Computing (HPDC-13), Juin 2004.

[Csc09]: **Computer-supported collaborative learning**, http://en.wikipedia.org/wiki/Computer-supported_collaborative_learning, janvier 2009.

[CSK+05]: A. Chervenak, R. Schuler, C. Kesselman, S. Koranda & B. Moe, "**Wide Area Data Replication for Scientific Collaborations**", in Proceedings of 6th IEEE/ACM Int'l Workshop on Grid Computing (Grid2005), November 2005.

[DBBO95]: P. Dillenbourg, M.Baker, A. Blaye & C. O'Malley, "**The evolution of research on collaborative learning**". In E. Spada & P. Reiman (Eds) Learning in Humans and Machine: Towards an interdisciplinary learning science. (Pp. 189-211) Oxford: Elsevier, 1995.

[Dic09]: **Data Intensive Computing**, http://dicomputing.pnl.gov/, Janvier 2009.

[DLJC06]: P. Dugénie, P. Lemoisson, C. Jonquet & M. Crubézy, "**The Grid Shared Desktop: a bootstrapping environment for collaboration**". Advanced Technology for Learning, Special issue on Collaborative Learning, Accepted for publication, 2006.

[DPC03]: P. Dillenbourg, C. Poirier & L. Carles, "**Communautés virtuelles d'apprentissage : e-jargon ou nouveau paradigme ?**" In A. Taurisson et A. Sentini. Pédagogiques.Net. Montréal, Presses, 2003.

[Dub03]: L. Dublin, "**If You Only Look Under the Street Lamps... Or Nine E-learning Myths** " The eLearning Developers' Journal, 1-7, 2003.

[Dug07]: P. Dugénie, "**Espace collaboratifs ubiquitaires, infrastructure à ressources distribuées**", PhD thesis, pp 32, Université des Sciences et Techniques du Languedoc, 2007.

[Exe09]: **Execution Management: Key Concepts**, http://www.globus.org/toolkit/docs/4.0/execution/key/, Janvier 2009.

[FCF+05]: I. Foster, K. Czajkowski, D. F. Ferguson, J. Frey, S. Graham, T. Maguire, D. Snelling & S. Tuecke. "Modeling **and managing state in distributed systems: The role of OGSI and WSRF**", Proceedings of the IEEE, 93(3):604-612, March 2005.

[FFM07]: M. Feller, I. Foster & S. Martin, "**GT4 GRAM: A Functionality and Performance Study**", TeraGrid Conference, 2007.

[FK04]: I. Foster & C. Kesselman, "**The Grid 2: Blueprint for a New Computing Infrastructure**", Second Edition. Morgan Kauffmann Publishers. USA, 2004.

[FK98]: I. Foster & C. Kesselman, "**The Globus Project: A Status Report**", Proc. IPPS/SPDP '98 Heterogeneous Computing Workshop, pp. 4-18, 1998.

[FK99a]: I. Foster, & C. Kesselman, eds., "**The Grid: Blueprint for a New Computing Infrastructure**", Morgan Kaufmann, San Francisco, 1999, http://www.mkp.com/grids/.

[FK99b]: I. Foster & C. Kesselman, "**Computational Grids**", Chapter 2 of "The Grid: Blueprint for a New Computing Infrastructure", Morgan-Kaufman, 1999.

[FKNT02]: I. Foster & C. Kesselman, J. M. Nick, & S. Tuecke, "**The Physiology of the Grid—An Open Grid Services Architecture for Distributed Systems Integration**", The Globus Alliance, (June 2002), http://www.globus.org/research/papers/ogsa.pdf.

[FKT01]: I. Foster, C. Kesselman & S. Tuecke, "**The Anatomy of the Grid Enabling Scalable Virtual Organizations",** International J. Supercomputer Applications, Vol. 15, No. 3, 2001.

[FL01]: F. Henri & K. Lundgren-Cayrol, "**Apprentissage collaboratif à distance : pour comprendre et concevoir les environnements d'apprentissage virtuels** ». Sainte-Foy (Québec, Canada) : Presses de l'Université du Québec, 2001, 181p.

[Fos02]: I. Foster, "**What is the Grid? A Three Point Checklist**", GRIDToday, 20 Juillet 2002.

[Fos05a]: I. Foster, "**Globus Toolkit Version 4: Software for Service-Oriented Systems**", IFIP International Conference on Network and Parallel Computing, Springer-Verlag LNCS 3779, pp 2-13, 2005.

[Fos05b]: I. Foster, "**A Globus Primer**", Août 2005, www.globus.org/toolkit/docs/4.0/key/GT4_Primer_0.6.pdf

[Fos06]: I. Foster, "**Globus Toolkit Version 4: Software for Service-Oriented Systems**", IFIP International Conference on Network and Parallel Computing, Springer-Verlag LNCS 3779, pp 2-13, 2006.

[Gen09]: **Genome@home**, http://genomeathome.stanford.edu, Janvier 2008.

[GKL+02]: L. Guy, P. Kunszt, E. Laure, H. Stockinger & K. Stockinger, "**Replica management in data grids**", Technical report, GGF5 Working Draft, July, 2002.

[Glo03]: **Globus Toolkit, l'architecture de référence**, http://www.01net.com/article/205795.html, Janvier 2008.

[GNA+04] : G. Gouarderes, R. Nkambou, C. Allison, J. Bradshaw, R. Buyya, S. A. Cerri, M. Eisenstadt, M. Liquiere, L. Razmerita, P. Ritrovato, D. D. Roure & R. Yatchou: Workshop on "**Grid Learning Services**". Intelligent Tutoring Systems 2004: 910.

[Gon08]: I. Gonon, "**Guide pratique du travail collaboratif en communautés virtuelles d'apprentissage**", avril 2008.

[Gri09]: **GRID'5000 Home page**, https://www.grid5000.fr/, janvier 2008.

[Gro09] : **le groupe**, www.ecogesam.ac-aix-marseille.fr/Resped/.../1-LeGroupe.doc, février 2009.

[Gru93]: T. R. Gruber, "**Formal ontology in conceptual analysis and knowledge representation**", Chapter: Towards principles for the design of ontologies used for knowledge sharing. Kluwer Academic Publishers, 1993.

[Gt309]: **GT 3.9.4 Information Services (MDS): Key Concepts**, http://www.globus.org/toolkit/docs/development/3.9.4/info/key/, Juin 2009.

[HB02]: J. Hagel & J.S. Brown, "**Service Grids: The Missing Link in Web Services**", White Paper, 2002.

[Hca09]: **How to Create Active and Effective Virtual Learning Communities**, http://www.brandon-hall.com/publications/virtuallearningcommunities/learningcommunities.shtml, Mars 2009.

[JBFT05]: B. Jacob, M. Brown, K. Fukui & N. Trivedi, "**Introduction to Grid Computing", IBM redbook**", December 2005.

[JEF04]: J.Joseph, M. Ernest & C. Fellenstein, "**Evolution of Grid computing architecture and Grid adoption models**", IBM SYSTEMS JOURNAL, VOL 43, NO 4, 2004.

[JP07]: A. Jukniute & G. Paulikas, "**SANDS VO: E-learning system architecture for Grid environment**". ISSN 1392 – 124X Information technology and control, 2007, Vol.36, No.1A.

[KMS04]. W. Kassem, A. Mounajed & N. Saadoun, "**état de l'art du E-learning**", Projet du module 4.1 : Management et NTIC, février 2004.

[LB05]: M. Li & M. Baker, "**The Grid: Core Technologies**". Wiley. ISBN 0-470-09417-6, 2005. http://coregridtechnologies.org/.

[Lea09]: **Learning management system**, http://en.wikipedia.org/wiki/Learning_management_system, fevrier 2009.

[Lec07]: B. Lecomte, "**Le choix d'un Learning Management System: une question institutionnelle**", IFRES-LabSET, ULg Colloque "Formation à Distance: nouveaux dispositifs et nécessaire accompagnement de tous les acteurs" Liège – 27 & 28, novembre 2007.

[Lef03]: A. Lefebvre, "**Computing On demand, rêve ou perspective?**", juillet 2003, http://www.journaldunet.com/solutions/0307/030715_chronique.shtml.

[Lms09]: **Les LMS/LCMS et leurs rôles dans le E-learning**, http://knol.google.com/k/jibril/les-lms-lcms-et-leurs-r%C3%B4les-dans-le/17euix69lwm49/3#, 2009.

[LTT08]: K.-C. Li, Y.-T. Tsai and C.-K. Tsai, "**Toward Development of Distance Learning Environment in the Grid**", Journal of Distance Education Technologies, 6(3), 45-57, Juillet-Septembre 2008.

[LYJS06]: Y. Li, S. Yang, J. Jiang & M. Shi, "**Build grid-enabled large-scale collaboration environment in E-learning Grid**", Expert Systems with Applications 31 (2006) 742–754.

[MAD05]: I. Madjarov, "**Des services web pour le e-Learning**", Octobre 2005. http://www.revue-eti.net/document.php?id=354

[Mds09]: **MDS4 and Project Deployments**, http://www.mcs.anl.gov/~schopf/Pubs/MDS4.PDJuly05.pdf , 2009.

[Ogf09]: **Open Grid Forum**, http://www.ggf.org, 2009.

[Ont09] : Ontologie (informatique), http://fr.wikipedia.org/wiki/Ontologie_(informatique), 2009.

[Par05] : P. Parrend, "**MDE et CSCW Groupware Travail Coopératif capillaire**", Juin 2005.

[PV03]: V. Pankratius & G. Vossen, "**Towards E-learning Grids: Using Grid Computing in Electronic Learning**" in Proc. IEEE Workshop on knowledge Grid and Grid Intelligence, Octobre 2003.

[SB94]: M. Scardamalia & C. Bereiter, "**Computer support for knowledge-building community**". Journal of the Learning Sciences, 3, 265-283, 1994.

[Set09]: Seti@home, http://setiathome.berkeley.edu, Mars 2008.

[Sew09]: **Les services Web en neuf questions**, http://www.journaldunet.com/solutions/0109/010924_faqWebservices.shtml, Mars 2008

[Sha07]: V. Shankar, "**CMS And LMS - A Comparison**", http://www.contentmanagementnews.com/contentmanagementnews-82 20070220CMSandLMSAComparison.html, Fevrier 2007.

[Tdg09]: **The Data GRID Project**, http://eu-datagrid.Web.cern.ch/eu-datagrid/, Janvier 2008.

[Tga09]: **The Globus Alliance**, http://www.globus.org, Ianvier 2009.

[Vli04]: E. van der Vlist, '**LE TRIPTYQUE SOAP/WSDL/UDDI**', Juin 2004.

[War05]: M Waris, "**Web Services Definition Language Version 2.0**", Novembre 2005.

[Wcg09]: **World community Grid**, http://www.worldcommunityGRID.org, Janvier 2008.

[Web09]: **Web Services**, http://www.softeam.fr/technologies_Web_services.php, Janvier 2008.

[Wel05]: V. Welch (ed.), "**Globus Toolkit Version 4 Grid Security Infrastructure: A Standards Perspective**", July 2005. http://www.globus.org/toolkit/docs/4.0/security/GT4-GSI-Overview.pdf.

[WP04] : M. Walckiers et T. De Praetere, "**L'apprentissage collaboratif en ligne, huit avantages qui en font un must**", Distances et savoirs 2004/1, Volume 2, p. 53-75.

[Wsa03]: **Web Services Architecture**, W3C Working Draft 8 August 2003. http://www.w3.org/TR/2003/WD-ws-arch-20030808/.

[Wsa09]: **Web Services Activity**, http://www.w3.org/2002/ws/, Mars 2008.

[YH05]: C.-T. YANG & H.-C. HO, "**An E-learning Platform Based on Grid Architecture**", Journal of information science and engineering 21, 911-928 (2005). A Standards Perspective", July 2005. http://www-unix.globus.org/toolkit/docs/4.0/security/GT4-GSI-Overview.pdf

Annexe

INSTALLATION DE GLOBUS (GT 4.6)

(Sous Linux Fedora core4)

Dans cette Annexe, nous allons exposer les principales étapes d'installation de l'intergiciel Globus dans sa version 4.0.6, sous le système LINUX Fedora core4. On mentionne que la courante installation peut être personnalisée selon les besoins des utilisateurs et leurs domaines d'application.

Une mise en œuvre d'un nœud de Grid, consiste à préparer d'abord l'environnement, l'initialiser, installer l'intergiciel, et enfin, configurer les services de cet intergiciel. Dans notre cadre d'étude, nous allons mettre en place un réseau LAN constitué de quatre machines, sur chacune est installé le système Linux Fedora core 4. nous avons utilisé Linux car c'est le système d'exploitation le plus approprié pour une installation complète de Globus, ainsi, il offre un environnement très fiable et sécurisé.

1. INSTALLATION D'UN PROGRAMME

Nous présentons ici quelques commandes qui sont nécessaires pour la procédure d'installation :

configure (./configure): Cette commande construit un fichier `Makefile utile pour la compilation d'un programme.`

make: Cette commande construit le programme. Autrement dit, **make** s'exécute, recherche la première cible dans le fichier `Makefile` et obéit aux instructions indiquées. Le résultat final attendu est la construction d'un programme exécutable.

make install: Cette commande invoque à nouveau **make**, qui recherche la cible `install` dans le `Makefile` et suit les instructions pour installer le programme.

L'éditeur vi : vi est un éditeur linux, il peut être à l'un des deux modes suivants : commande et édition. Le passage du premier au second se fait par la touche i (insert), le passage inverse se fait par la touche esc. On peut éditer en mode édition et enregistrer no modification en mode commande par (':wq' pour écrire dans le fichier et sortir) ou par (' :q' pour quitter sans enregistrer les modification). Afin de pouvoir modifier un fichier on doit avoir l'autorisation en écriture.
Exemple « vi /fichier ».

Notamment, ils existent d'autres éditeurs comme vim et nano.

2. MISE EN PLACE DU RESEAU

2.1. Quelques caractéristiques des machines utilisées

- Fabriquant : Intel.
- Processeur: Intel(R) Pentium(R) 4 CPU 3.00GHZ (2CPUs).
- Mémoire: 1 GB RAM.

2.2. Installation du système linux

La version Linux utilisée est Fedora 8 Core version 4, en faisant attention aux points suivants:

- Type d'installation poste de travail (WorkStation).
- Désactivation de pare-feu pour ne pas gêner le fonctionnement des services de l'intergitiel.
- Date et heure doivent être réglées pour chaque machine, cette condition est indispensable dans la phase de signature de certificat pour les autres machines afin de vérifier la validité du proxy.

2.3. Configuration du réseau

Lors de l'installation de linux, nous avons attribué un nom et une adresse IP à chaque noeud de la grille qui vont servir par la suite à la configuration de l'intergiciel globus. Chaque nom du hôte doit avoir la forme suivante:'nom_machine.nom_domaine' (une exigence de l'intergiciel globus toolkit).

Dans cette illustration la grille est configurée de la manière suivante:

Nom de la machine	Adresse IP	Description
Poste.lri.net	192.168.0.5	Machine client/serveur
Poste2.lri.net	192.168.0.103	Machine client/serveur
Poste3.lri.net	192.168.0.104	Machine client/serveur
Poste4.lri.net	192.168.0.101	Machine client/serveur propriétaire du certificat

2.4. Outils nécessaires

- Apache Ant : c'est un exécuteur de tâches il permet le déploiement des programmes 'déploiement de services)

- JDK : nécessaire pour la compilation du code de globus car la plus grande partie est écrite en java.
- PostgresSQL : système de gestion de base de donnée relationnelle fonctionnant sur des systèmes UNIX. Il est composé de deux parties :
  - Partie serveur : c'est la partie fonctionnant sur la machine hébergeant la base de données capable de traiter les requêtes des clients
  - Partie client : cette partie est installée sur les postes client. Les clients interrogent le serveur de base de données par des requêtes SQL.
- Globus Toolkit : Une boite à outils qui contient des fichiers et des Ordonnanceurs et d'autres outils nécessaires pour faire le calcul distribué, assurer la sécurité …

3. Préparation de l'installation de globus toolkit 4.0.6

3.1. Création des comptes utilisateurs

Afin d'installer globus et pouvoir tester son bon fonctionnement, nous devons créer trois types d'utilisateurs: l'administrateur du système de ce poste qui permet le lancement et l'arrêt du container, c'est un utilisateur non privilégié par apport à **root** qui sera créé automatiquement lors de l'installation du système Linux et qui possède tous les droits, l'utilisateur '**globus**' qui servira à l'installation de l'intergiciel globus et un utilisateur simple '**user**'.

La création des utilisateurs peut se faire lors de l'installation de linux ou à l'aide de la commande '**adduser**', ou même avec le gestionnaire des groupes et utilisateurs sous Fedora.

3.2. Création des répertoires d'installation:

Nous créons deux répertoires dans chaque noeud de la grille, sous le répertoire '/usr/local', un pour globus et l'autre pour les outils (jdk, ant...).

- Création d'un répertoire d'installation globus-4.0.6 sous ***'/usr/local'***

```
[root@poste4]# mkdir /usr/local/globus-4.0.6
[root@poste4]# chown globus:globus /usr/local/globus-4.0.6
```

- Création du répertoire d'installation des outils '***outils***' sous ***'/usr/local',*** de la même façon.

```
[root@poste4]# mkdir /usr/local/outils
```

Dans le répertoire ***'outils'*** déjà crée, on copie les outils : Java, Apache-ant et Postgresql.

3.3. Installation des outils apache Java, Apache-ant et Postgresql

- **Installation de Java**

Après le téléchargement du jdk-1_5_0_14, nous lançons son installation dans le répertoire '*outils*' par la commande d'installation suivante :

```
[root@poste4]# cd /usr/local/outils
[root@poste4 outils]# ./jdk-1_5_0_14-nb-6_0-linux.sh
```

Après l'installation, nous devons ajouter à la variable d'environnement 'path' le chemin de jdk. Nous pouvons procéder de deux manières distinctes : dans la première nous modifions le fichier /etc/profile.d et dans la seconde, nous créons un fichier jdk.sh sous /etc/profile.d dont le contenu est le suivant:



Une vue du fichier */etc/profile.d/jdk.sh*

Remarque : Il faut supprimer le lien vers l'ancienne version de Java représenté par le fichier raccourci 'java' trouvée sous le chemin */usr/bin*, et nous pouvons tester la réussite de l'installation du JDK en tapant la commande :

```
[root@poste4]# java -version
```

Le résultat donc sera :

```
java version "1.5.0_14"
Java(TM) 2 Runtime Environment, Standard Edition (build 1.5.0_14-b03)
Java HotSpot(TM) Client VM (build 1.5.0_14-b03, mixed mode, sharing)
```

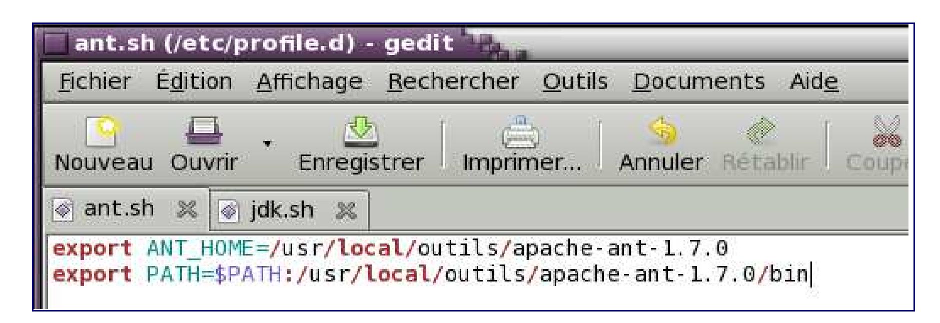
- **Installation de Apache ant**

Apache ant est un exécuteur de tâches permettant de compiler et déployer des programmes, après le 'téléchargement de la version de Apache ant 'appache-ant-1.7.0-bin.tar.gz', on l'installe dans le répertoire '/usr/local/outils'.

L'installation de Apache ant consiste à le décompresser, et ajouter à la variable d'environnement 'path' le chemin de ant en créant un fichier ant.sh dans '/etc/profile.d'. La décompression se fait par :

```
[root@poste4 outils]# tar -xzf appache-ant-1.7.0-bin.tar.gz
```

Un aperçu du fichier 'ant.sh' est illustré par la figure suivante:



Une vue du fichier _/etc/profile.d/ant.sh_

Nous devons tester si *apache ant* est bien installé en utilisant la commande suivante:

```
[root@poste4 ~]# ant – version
```

Le résultat sera comme suit :

```
Apache Ant version 1.7.0 compiled on July 13 2008
```

- **Installation de postgresql**

Afin d'installer Postgresql, on télécharge le fichier postgresql-8.2.6.tar.gz, on le décompresse dans '/usr/local/' :

```
[root@poste4 outils]# tar -xzf postgresql-8.2.6.tar.gz -C /usr/local/
```

On lance la commande suivante:

```
[root@poste4 postgresql-8.2.6]# ./configure -without-readline -without-zlib
```

On lance l'installation:

```
[root@poste4 postgresql-8.2.6] # make
[root@poste4 postgresql-8.2.6] # make install
```

Ensuite, nous créons un utilisateur postgres et nous créons le répertoire '*pgsql/data*'.

```
[root@poste4] # adduser postgres
[root@poste4] # mkdir /usr/local/pgsql/data
[root@poste4] # chown postgres /usr/local/pgsql/data
```

En tant qu'utilisateur 'postgres', on exécute la commande suivante :

```
[postgres@poste4] $/usr/local/pgsql/bin/initdb -D /usr/local/pgsql/data |
tee initdb.log
```

On lance le serveur de base de données :

```
[postgres@poste4]$ cd /usr/local
```

```
[postgres@poste4 local]$ pgsql/bin/postmaster -D pgsql/data
```

Le résultat sera:

```
LOG:  database system was interrupted at 2008-07-13 13:09:54 CET
LOG:  checkpoint record is at 0/42E904
LOG:  redo record is at 0/42E904; undo record is at 0/0; shutdown FALSE
LOG:  next transaction ID: 0/622; next OID: 24579
LOG:  next MultiXactId: 1; next MultiXactOffset: 0
LOG:  database system was not properly shut down; automatic recovery in progress
LOG:  record with zero length at 0/42E94C
LOG:  redo is not required
LOG:  database system is ready
```

Pour vérifier si la base de données est bien installée, nous créons une base de données test comme suit :

```
[postgres@poste4]$ /usr/local/pgsql/bin/createdb test
```

Le résultat de cette commande est :

```
CREATE DATABASE
```

Nous testons la création de la base par cette commande :

```
[postgres@poste4]$  /usr/local/pgsql/bin/psql test
```

Le résultat est :

```
Welcome to psql 8.2.6, the PostgreSQL interactive terminal.
Type:  \copyright for distribution terms
       \h for help with SQL commands
       \? for help with psql commands
       \g or terminate with semicolon to execute query
       \q to quit
test=#
```

4. L’installation de Globus Toolkit:

4.1. Lancement du script d’installation

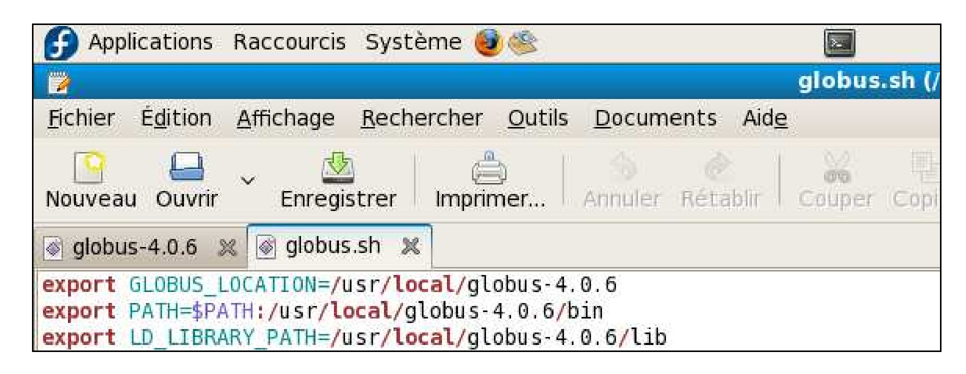
La version téléchargée de globus toolkit est ‘gt4.0.6-x86_fc_4-installer.tar.gz’. Nous commençons par la décompression du fichier dans ‘/usr/local/globus-4.0.6’, copier les fichiers décompressés dans le même répertoire, ajouter la variable $GLOBUS_LOCATION au path et ensuite lancer l'opération d'installation :

a) Décompresser et copier :

```
[root@poste4 ~]# tar -xzf /usr/local/gt4.0.6-x86_fc_4-installer.tar.gz -C /usr/local/globus-4.0.6/

[root@poste4 ~]# cp -r /usr/local/globus-4.0.6/gt4.0.6-x86_fc_4-installer/* /usr/local/globus-4.0.6/
```

b) Aperçu sur le fichier '/etc/profile.d/globus.sh'



Une vue du fichier ***/etc/profile.d/globus.sh***

c) Désigner l'utilisateur et le groupe propriétaire des fichiers, cette étape permet d'affecter à l'utilisateur globus l'ensemble des fichiers décompressés utiles pour l'installation de l'intergiciel. Avec la commande qui suit, l'installation peut être faite sous l'utilisateur globus:

```
[root@poste4 ~]# chown -R globus:globus /usr/local/globus-4.0.6/
```

d) L'installation de globus toolkit doit être faite sous l'utilisateur globus, nous créons un fichier `Makefile` avec la commande *./configure :*

```
[globus@poste4 ~]$ cd /usr/local/globus-4.0.6

[globus@poste4 globus-4.0.6]$ ./configure -prefix=$GLOBUS_LOCATION -enable-prewsmd -enable-drs
```

Le résultat sera :

```
checking for javac... /usr/local/outils/jdk1.5.0_14/bin/javac
checking for ant... /usr/local/outils/apache-ant-1.7.0/bin/ant
configure: creating ./config.status
config.status: creating Makefile
```

Les caractéristiques suivantes sont optionnelles et sont toutes désactivées par défaut.

-enable-prewsmd : permet de construire la base de pre-webservices mds.

-enable-wsgram-condor : permet de construire l'interface de GRAM Condor scheduler.

-enable-wsgram-lsf: permet de construire l'interface de GRAM LSF scheduler.

-enable-i18n: permet d'activer l'internationalisation.

-enable-drs: permet d'activer le Service de Réplication de données

e) Construire le programme exécutable avec la commande make, nous utilisons *tee* pour garder la trace de l'exécution des commandes d'installation, afin d'aider à localiser les erreurs qui peuvent se produire durant l'installation :

```
[globus@poste4 globus-4.0.6 ]$ make |tee build.log
```

Une partie du résultat de cette commande sera comme suit :

```
tar -C /usr/local/globus-4.0.6 -xzf binary-trees/globus_wsrf_replicator_common_schema-*/*.tar.gz
tar -C /usr/local/globus-4.0.6 -xzf binary-trees/globus_wsrf_replicator_common_java-*/*.tar.gz
tar -C /usr/local/globus-4.0.6 -xzf binary-trees/globus_wsrf_replicator_service_java-*/*.tar.gz
tar -C /usr/local/globus-4.0.6 -xzf binary-trees/globus_wsrf_replicator_client_java-*/*.tar.gz
echo "Your build completed successfully.  Please run make install."
Your build completed successfully.  Please run make install.
```

f) Lancer l'installation de globus avec la commande *make install* :

```
[globus@poste4 globus-4.0.6 ]$ make install|tee install.log
```

Une partie du résultat est la suivante :

```
running /usr/local/globus-4.0.6/setup/globus/setup-globus-job-manager-fork..[ Changing to /usr/local/globus-4.0.6/setup/globus ]
find-fork-tools: WARNING: "Cannot locate mpiexec"
find-fork-tools: WARNING: "Cannot locate mpirun"
checking for mpiexec... no
checking for mpirun... no
find-fork-tools: creating ./config.status
config.status: creating fork.pm
..Done
```

4.2. L'installation de l'Autorité de Certification (CA)

L'installation de l'autorité de certification doit se faire sur une seule machine qu'on appelle serveur de certificat, cette étape est très critique pour le reste des étapes d'installation et de configuration de différents services de l'intergiciel.

Le script d'installation de l'AC est placé lors de l'étape précédente, l'installation du serveur du certificat doit être faite sous l'utilisateur globus.

4.2.1. Exécution du script d'installation

a) Nous rajoutons au fichier **«hosts»** qui se trouve dans le répertoire ***'/etc/'*** le nom de la machine et l'adresse IP correspondante de la façon suivante:

192.168.0.101 poste4.lri.net poste4

b) Sous root, nous créons le répertoire *'/etc/grid-security'* qui va contenir le certificat du hôte. Après la création, nous affectons le répertoire à l'utilisateur globus :

```
[root@poste4]# mkdir /etc/grid-security
[root@poste4]# chown globus:globus /etc/grid-security
```

c) Sous globus, nous lançons l'exécution du script :

```
[globus@poste4]$ $GLOBUS_LOCATION/setup/globus/setup-simple-ca
```

Nous confirmons l'exécution en tapant 'y', puis nous remplissons les champs demandés durant l'installation (adresse mail, durée d'expiration du certificat, mot de passe). Une partie du résultat est la suivante:

```
WARNING: GPT_LOCATION not set, assuming:
        GPT_LOCATION=/usr/local/globus-4.0.6
    C e r t i f i c a t e   A u t h o r i t y   S e t u p
This script will setup a Certificate Authority for signing Globus
users certificates.  It will also generate a simple CA package
that can be distributed to the users of the CA.
The CA information about the certificates it distributes will
be kept in:
/home/globus/.globus/simpleCA/
The unique subject name for this CA is:
cn=Globus Simple CA, ou=simpleCA-poste4.lri.net, ou=GlobusTest, o=Grid
Do you want to keep this as the CA subject (y/n) [y]:y
Enter the email of the CA (this is the email where certificate
requests will be sent to be signed by the CA):ahmed.dib@lri-annaba.net
The CA certificate has an expiration date. Keep in mind that
once the CA certificate has expired, all the certificates
signed by that CA become invalid.  A CA should regenerate
the CA certificate and start re-issuing ca-setup packages
before the actual CA certificate expires.  This can be done
by re-running this setup script.  Enter the number of DAYS
the CA certificate should last before it expires.
[default: 5 years (1825 days)]:1825
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
creating CA config package...done.
A self-signed certificate has been generated
for the Certificate Authority with the subject:
/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/CN=Globus Simple CA
```

```
If this is invalid, rerun this script
/usr/local/globus-4.0.6/setup/globus/setup-simple-ca
and enter the appropriate fields.
-------------------------------------------------------------------
The private key of the CA is stored in /home/globus/.globus/simpleCA//private/cakey.pem
The public CA certificate is stored in /home/globus/.globus/simpleCA//cacert.pem
The distribution package built for this CA is stored in
/home/globus/.globus/simpleCA//globus_simple_ca_637244ab_setup-0.19.tar.gz
This file must be distributed to any host wishing to request
certificates from this CA.
CA setup complete.
The following commands will now be run to setup the security
configuration files for this CA:
$GLOBUS_LOCATION/sbin/gpt-build /home/globus/.globus/simpleCA//globus_simple_ca_637244ab_setup-0.19.tar.gz
$GLOBUS_LOCATION/sbin/gpt-postinstall
-------------------------------------------------------------------
setup-ssl-utils: Configuring ssl-utils package
Running setup-ssl-utils-sh-scripts...
***************************************************************************
Note: To complete setup of the GSI software you need to run the
following script as root to configure your security configuration
directory:
/usr/local/globus-4.0.6/setup/globus_simple_ca_637244ab_setup/setup-gsi
For further information on using the setup-gsi script, use the -help
option.  The -default option sets this security configuration to be
the default, and -nonroot can be used on systems where root access is
not available.
***************************************************************************
setup-ssl-utils: Complete
```

4.2.2. Installation du service GSI

Pour lancer l'installation de gsi on tape la commande suivante :

```
[root@poste4]# /usr/local/globus-4.0.6/setup/globus_simple_ca_637244ab_setup/setup-gsi
```

Le résultat est le suivant :

```
setup-gsi: Configuring GSI security
Making trusted certs directory: /etc/grid-security/certificates/
mkdir /etc/grid-security/certificates/
Installing /etc/grid-security/certificates//grid-security.conf.637244ab...
```

```
Running grid-security-config...
Installing Globus CA certificate into trusted CA certificate directory...
Installing Globus CA signing policy into trusted CA certificate directory...
setup-gsi: Complete
[globus@poste4 globus-4.0.6]$ grid-cert-request -ca
nondefaultca=true
The available CA configurations installed on this host are:
1)  637244ab - /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/CN=Globus Simple CA
Enter the index number of the CA you want to sign your cert request: 1
Using CA: 637244ab - /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/CN=Globus Simple CA
A certificate request and private key is being created.
You will be asked to enter a PEM pass phrase.
This pass phrase is akin to your account password,
and is used to protect your key file.
If you forget your pass phrase, you will need to
obtain a new certificate.
Generating a 1024 bit RSA private key
......++++++
....................................................++++++
writing new private key to '/home/globus/.globus/userkey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Level 0 Organization [Grid]:Level 0 Organizational Unit [GlobusTest]:Level 1 Organizational Unit [simpleCA-poste4.lri.net]:Level 2 Organizational Unit [lri.net]:Name (e.g., John M. Smith) []:
A private key and a certificate request has been generated with the subject:
/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=globus
If the CN=globus is not appropriate, rerun this
script with the -force -cn "Common Name" options.
Your private key is stored in /home/globus/.globus/userkey.pem
Your request is stored in /home/globus/.globus/usercert_request.pem
Please e-mail the request to the Globus Simple CA ahmed.dib@lri-annaba.net
You may use a command similar to the following:
```

```
cat /home/globus/.globus/usercert_request.pem | mail ahmed.dib@lri-annaba.net
Only use the above if this machine can send AND receive e-mail. if not, please
mail using some other method.
Your certificate will be mailed to you within two working days.
If you receive no response, contact Globus Simple CA at ahmed.dib@lri-annaba.net
```

4.2.3. Génération du certificat pour le hôte

a) Demande certificat pour un nœud hôte *'host certificates'*

Se fait par la commande suivante sous root:

```
[root@poste4]# grid-cert-request -host poste4.lri.net
```

Cette commande nous donne comme résultat trois fichiers en sortie dans '/etc/grid-security', nous remarquons bien que dans le résultat, nous trouvons que le fichier 'hostcert.pem' qui va porter le certificat valide et de taille zéro, en faite, ce fichier va contenir le résultat de signature du fichier portant la demande de certificat 'hostcert_request.pem' :

```
-rw-r--r-- 1 root   root      0    jui 20 22:46 hostcert.pem
-rw-r--r-- 1 root   root   1396    jui 20 22:46 hostcert_request.pem
-r-------- 1   root   root    887    jui 20 22:46 hostkey.pem
```

b) Signature du certificat «host certificates»

La signature du certificat se fait par l'utilisateur globus, mais le fichier à signer hostcert_request.pem appartient à root, nous devons faire attention aux différents droits des utilisateurs sur les fichiers.

Nous procédons à la signature du certificat de l'hôte :

```
[root@poste4~]# cp /etc/grid-security/hostcert_request.pem /tmp/

[globus@poste4~]$ grid-ca-sign -in /tmp/hostcert_request.pem -out
/tmp/hostsigend.pem
```

Le résultat sera :

```
To sign the request
please enter the password for the CA key:
The new signed certificate is at: /home/globus/.globus/simpleCA//newcerts/01.pem
```

Nous déplaçons le fichier contenant la clé signée 'hostsigned.pem' vers le fichier qui va contenir le certificat signé de l'hôte 'hostcert.pem':

```
[root@poste4 .globus]# cp /tmp/hostsigend.pem /etc/grid-security/

[root@poste4 .globus]# mv /etc/grid-security/hostsigend.pem /etc/grid-
security/hostcert.pem
```

```
mv: écraser `/etc/grid-security/hostcert.pem'?y
```

Nous vérifions le contenu de '/etc/grid-security'

```
-rw-r--r-- 1 root   root   2682 jui 20 22:55 hostcert.pem
-rw-r--r-- 1 root   root   1407 jui 20 22:46 hostcert_request.pem
-r-------- 1 root   root    891 jui 20 22:46 hostkey.pem
```

4.2.4. Génération du certificat pour l'utilisateur globus

a) Demande du certificat de l'utilisateur globus

Avec la commande suivante:

```
[globus@poste4 ]$ grid-cert-request -ca
```

Le résultat de cette commande est comme suit :

```
nondefaultca=true
The available CA configurations installed on this host are:
1)  637244ab - /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/CN=Globus Simple CA
Enter the index number of the CA you want to sign your cert request: 1
Using CA: 637244ab - /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/CN=Globus Simple CA
A certificate request and private key is being created.
You will be asked to enter a PEM pass phrase.
This pass phrase is akin to your account password,
and is used to protect your key file.
If you forget your pass phrase, you will need to
obtain a new certificate.
Generating a 1024 bit RSA private key
......++++++
...................................................++++++
writing new private key to '/home/globus/.globus/userkey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Level 0 Organization [Grid]:Level 0 Organizational Unit [GlobusTest]:Level 1 Organizational Unit [simpleCA-poste4.lri.net]:Level 2 Organizational Unit [lri.net]:Name (e.g., John M. Smith) []:
```

```
A private key and a certificate request has been generated with the subject:
/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=globus
If the CN=globus is not appropriate, rerun this
script with the -force -cn "Common Name" options.
Your private key is stored in /home/globus/.globus/userkey.pem
Your request is stored in /home/globus/.globus/usercert_request.pem
Please e-mail the request to the Globus Simple CA ahmed.dib@lri-annaba.net
You may use a command similar to the following:
  cat /home/globus/.globus/usercert_request.pem | mail ahmed.dib@lri-annaba.net
Only use the above if this machine can send AND receive e-mail. if not, please
mail using some other method.
Your certificate will be mailed to you within two working days.
If you receive no response, contact Globus Simple CA at ahmed.dib@lri-annaba.net
```

Cette commande nous génère trois fichiers en sortie sous '/home/globus/.globus/':

```
userkey.pem    usercert_request.pem    usercert.pem
```

b) Signature du certificat de l'utilisateur globus

Après génération du fichier 'usercert_request.pem', nous devons le signer avec la commande suivante.

```
[globus@poste4 globus-4.0.6]$ grid-ca-sign -in
/home/globus/.globus/usercert_request.pem -out
/home/globus/.globus/usercert.pem
```

Le résultat est:

```
To sign the request
please enter the password for the CA key:
The new signed certificate is at: /home/globus/.globus/simpleCA//newcerts/02.pem
```

Nous devons vérifier que les fichiers du certificat de globus ont les droits appropriés :

```
-rw-r--r-- 1 globus globus 2689 jui 20 22:52 usercert.pem
-rw-r--r-- 1 globus globus 1415 jui 20 22:47 usercert_request.pem
-r-------- 1 globus globus  963 jui 20 22:47 userkey.pem
```

4.2.5. Génération du certificat pour l'utilisateur user

a) Demande du certificat de l'utilisateur 'user'

Partant du même principe, nous lançons la commande suivante sous l'utilisateur 'user':

```
[user@poste4 ]$ grid-cert-request –ca -force
```

Le résultat sera :

```
nondefaultca=true
```

```
The available CA configurations installed on this host are:
1)  637244ab - /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/CN=Globus Simple CA
Enter the index number of the CA you want to sign your cert request: 1
Using CA: 637244ab - /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/CN=Globus Simple CA
 /home/user/.globus/usercert_request.pem already exists
 /home/user/.globus/usercert.pem already exists/bin/chmod: modification des permissions de
`/home/user/.globus/usercert.pem': Opération non permise
  /home/user/.globus/userkey.pem already exists
A certificate request and private key is being created.
You will be asked to enter a PEM pass phrase.
This pass phrase is akin to your account password,
and is used to protect your key file.
If you forget your pass phrase, you will need to
obtain a new certificate.
Generating a 1024 bit RSA private key
.............................................................................................++++++
......++++++
writing new private key to '/home/user/.globus/userkey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Level 0 Organization [Grid]:Level 0 Organizational Unit [GlobusTest]:Level 1 Organizational Unit [simpleCA-
poste4.lri.net]:Level 2 Organizational Unit [lri.net]:Name (e.g., John M. Smith) []:


A private key and a certificate request has been generated with the subject:
/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=user
If the CN=user is not appropriate, rerun this
script with the -force -cn "Common Name" options.
Your private key is stored in /home/user/.globus/userkey.pem
Your request is stored in /home/user/.globus/usercert_request.pem
Please e-mail the request to the Globus Simple CA ahmed.dib@lri-annaba.net
You may use a command similar to the following:
cat /home/user/.globus/usercert_request.pem | mail ahmed.dib@lri-annaba.net
```

```
Only use the above if this machine can send AND receive e-mail. if not, please
mail using some other method.
Your certificate will be mailed to you within two working days.
If you receive no response, contact Globus Simple CA at ahmed.dib@lri-annaba.net
```

b) signature du certificat de l'utilisateur user

Nous signons le fichier ‘usercert_request.pem’ avec la commande suivante sous globus.

```
[user@poste4 ~]$ cp /home/user/.globus/usercert_request.pem /tmp/user_usercertrequest.pem
[globus@poste4 ~]$ grid-ca-sign -in /tmp/user_usercertrequest.pem -out /tmp/usersigned2.pem
[user@poste4 ~]$ cp /tmp/usersigned2.pem /home/user/.globus/
[user@poste4 ~]$ mv /home/user/.globus/usersigned2.pem /home/user/.globus/usercert.pem
[root@poste4 ~]# chmod 644 /home/user/.globus/usercert.pem
```

Nous vérifions le contenu de ‘/home/user/.globus/’ :

```
-rw-r--r-- 1 user user 2683 jui 20 23:04 usercert.pem
-rw-r--r-- 1 user user 1407 jui 20 22:48 usercert_request.pem
-r-------- 1 user user  963 jui 20 22:48 userkey.pem
```

4.2.6. Création du certificat du 'container'

Il est pratiquement nécessaire pour le lancement des services web, nous procédons comme suit :

```
[root@poste4]# cp /etc/grid-security/hostcert.pem /etc/grid-security/containercert.pem
[root@poste4]# cp /etc/grid-security/hostkey.pem /etc/grid-security/containerkey.pem
[root@poste4]# chown globus:globus /etc/grid-security/container*.pem
```

4.2.7. Ajout des autorisations

a) Ajout des autorisations pour l’utilisateur globus

L’ajout des autorisations se fait par la création du fichier ‘grid-mapfile’ qui garantie une communication sécurisée entre les différents noeuds de la grille, sa création peut être faite en utilisant l'éditeur vi, sous le répertoire grid-security, nous lançons la commande:

```
[root@ poste4]# vi /etc/grid-security/grid-mapfile
```

Nous ajoutons dans le fichier créé une entrée correspondante à chaque utilisateur (globus et user), les entrées à ajouter sont composées du sujet et le propriétaire du certificat, pour récupérer ces derniers, nous procédons comme suit pour chaque utilisateur:

```
[globus@poste4]$ grid-cert-info -subject
/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=globus
```

```
[globus@poste4]$ whoami
```

```
globus
```

Sous root, nous ajoutons une entrée correspondante au certificat de l'utilisateur globus dans le fichier grid-mapfile :

```
[root@poste4]# /usr/local/globus-4.0.6/sbin/grid-mapfile-add-entry -dn
"/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=globus" -ln
globus
```

Le résultat sera :

```
Modifying /etc/grid-security/grid-mapfile ...
New entry:
"/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=globus" globus
(1) entry added
```

b) Ajout des autorisations pour l'utilisateur user

Nous procédons de la même manière :

Pour récupérer le sujet du certificat et son propriétaire, nous exécutons :

```
[user@poste4 ~]$ grid-cert-info -subject
```

Le résultat sera :

```
/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=user
```

Après, nous lançons cette commande:

```
[user@poste4 ~]$ whoami
```

Résultat:

```
user
```

Sous root, nous ajoutons une entrée correspondante au certificat de l'utilisateur globus dans le fichier grid-mapfile :

```
[root@poste4 ~]# /usr/local/globus-4.0.6/sbin/grid-mapfile-add-entry -dn "
/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=user" -ln
user
```

Résultat:

```
Modifying /etc/grid-security/grid-mapfile ...
New entry:
" /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=user" user
(1) entry added
```

4.2.8. Vérification des certificats des utilisateurs

Pour la vérification du certificat de l'utilisateur globus, nous utilisons la commande suivante:

```
[globus@ poste4/]$ grid-proxy-init -debug –verify
```

Le résultat est le suivant:

```
User Cert File: /home/globus/.globus/usercert.pem
User Key File: /home/globus/.globus/userkey.pem
Trusted CA Cert Dir: /etc/grid-security/certificates
Output File: /tmp/x509up_u500
Your identity: /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=globus
Enter GRID pass phrase for this identity:
Creating proxy ..........++++++++++++
...........++++++++++++
 Done
Proxy Verify OK
Your proxy is valid until: Mon Jul 21 11:12:10 2008
```

De même, sous l'utilisateur user :

```
[user@ poste4/]$ grid-proxy-init -debug -verify
```

```
User Cert File: /home/user/.globus/usercert.pem
User Key File: /home/user/.globus/userkey.pem
Trusted CA Cert Dir: /etc/grid-security/certificates
Output File: /tmp/x509up_u501
Your identity: /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=user
Enter GRID pass phrase for this identity:
Creating proxy .......................++++++++++++
...++++++++++++
 Done
Proxy Verify OK
Your proxy is valid until: Mon Jul 21 11:12:19 2008
```

a) Installation des certificats pour plusieurs machines

Le certificat sera installé sur les machines autres que celle où nous avons installé le certificat d'autorité.

A partir de la machine 4, nous copions le fichier 'globus_simple_ca_637244ab_setup-0.19.tar.gz.' qui se trouve sous «/home/globus/.globus/simpleCA» sur les machines où nous allons installer le certificat sous «home/globus/», puis nous lançons la commande suivante sous globus,

nous prenons comme exemple l'installation du certificat sur la machine 'poste' :

```
[root@poste4 sbin]# scp
/home/globus/.globus/simpleCA/globus_simple_ca_637244ab_setup-0.19.tar.gz
root@poste.lri.net/home/globus/

[globus@poste ~]$ $GLOBUS_LOCATION/sbin/gpt-build
globus_simple_ca_637244ab_setup-0.19.tar.gz gcc32dbg
```

Nous obtenons :

```
gpt-build ====> Changing to /home/globus/BUILD/globus_core-4.30/
gpt-build ====> BUILDING FLAVOR gcc32dbg
gpt-build ====> Changing to /home/globus/BUILD
gpt-build ====> REMOVING empty package globus_core-gcc32dbg-pgm_static
gpt-build ====> REMOVING empty package globus_core-noflavor-doc
gpt-build ====> CHECKING BUILD DEPENDENCIES FOR globus_simple_ca_637244ab_setup
gpt-build ====> Changing to /home/globus/BUILD/globus_simple_ca_637244ab_setup-0.19/
gpt-build ====> BUILDING globus_simple_ca_637244ab_setup
gpt-build ====> Changing to /home/globus/BUILD
gpt-build ====> REMOVING empty package globus_simple_ca_637244ab_setup-noflavor-data
gpt-build ====> REMOVING empty package globus_simple_ca_637244ab_setup-noflavor-dev
gpt-build ====> REMOVING empty package globus_simple_ca_637244ab_setup-noflavor-doc
gpt-build ====> REMOVING empty package globus_simple_ca_637244ab_setup-noflavor-pgm_static
gpt-build ====> REMOVING empty package globus_simple_ca_637244ab_setup-noflavor-rtl
```

Puis la commande:

```
[globus@poste]$ $GLOBUS_LOCATION/sbin/gpt-postinstall
```

Le résultat sera:

```
running /usr/local/globus-4.0.6//setup/globus/./setup-ssl-utils.637244ab..[ Changing to /usr/local/globus-4.0.6/setup/globus/. ]
setup-ssl-utils: Configuring ssl-utils package
Running setup-ssl-utils-sh-scripts...
***************************************************************************
Note: To complete setup of the GSI software you need to run the
following script as root to configure your security configuration
directory:
/usr/local/globus-4.0.6//setup/globus_simple_ca_637244ab_setup/setup-gsi
For further information on using the setup-gsi script, use the -help
option.  The -default option sets this security configuration to be
the default, and -nonroot can be used on systems where root access is
not available.
```

```
*****************************************************************************
setup-ssl-utils: Complete
..Done
WARNING: The following packages were not set up correctly:
        globus_simple_ca_637244ab_setup-noflavor-pgm
Check the package documentation or run postinstall -verbose to see what happened
```

Maintenant nous installons le gsi par la commande suivante:

```
[root@poste]#$GLOBUS_LOCATION/setup/globus_simple_ca_637244ab_setup/setup-gsi -default
```

Le résultat :

```
bash: /usr/local/globus-4.0.6/setup/globus_simple_ca_637244ab_setup/etup-gsi: Aucun fichier ou répertoire de ce type
[root@poste ~]# /usr/local/globus-4.0.6/setup/globus_simple_ca_637244ab_setup/setup-gsi -default
setup-gsi: Configuring GSI security
Making /etc/grid-security...
mkdir /etc/grid-security
Making trusted certs directory: /etc/grid-security/certificates/
mkdir /etc/grid-security/certificates/
Installing /etc/grid-security/certificates//grid-security.conf.637244ab...
Running grid-security-config...
Installing Globus CA certificate into trusted CA certificate directory...
Installing Globus CA signing policy into trusted CA certificate directory...
setup-gsi: Complete
```

Nous ajoutons au fichier '*/etc/hosts*' les adresses IP ainsi les noms des autres hôtes reliés par la grille. Nous ouvrons le fichier avec l'éditeur vi :

```
[root@poste]# vi /etc/hosts
```

Et nous ajoutons ce qui suit :

192.168.0.103 poste2.lri.net

192.168.0.104 poste3.lri.net

192.168.0.101 poste4.lri.net

Puis, nous lançons la demande du certificat pour l'hôte du 'poste' :

```
[root@poste ~]# grid-cert-request -host poste.lri.net
```

Le résultat :

```
Generating a 1024 bit RSA private key
```

```
..........++++++
.......++++++
writing new private key to '/etc/grid-security/hostkey.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Level 0 Organization [Grid]:Level 0 Organizational Unit [GlobusTest]:Level 1 Organizational Unit [simpleCA-poste4.lri.net]:Name (e.g., John M. Smith) []:
 A private host key and a certificate request has been generated
with the subject:
/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/CN=host/poste.lri.net
----------------------------------------------------------
The private key is stored in /etc/grid-security/hostkey.pem
The request is stored in /etc/grid-security/hostcert_request.pem
Please e-mail the request to the Globus Simple CA ahmed.dib@lri-annaba.net
You may use a command similar to the following:
 cat /etc/grid-security/hostcert_request.pem | mail ahmed.dib@lri-annaba.net
Only use the above if this machine can send AND receive e-mail. if not, please
mail using some other method.
Your certificate will be mailed to you within two working days.
If you receive no response, contact Globus Simple CA at ahmed.dib@lri-annaba.net
```

Dans le répertoire /etc/*grid-security*, trois fichiers sont créés :

```
-rw-r--r-- 1 root root    0 jui 21 10:20 hostcert.pem
-rw-r--r-- 1 root root 1406 jui 21 10:20 hostcert_request.pem
-r-------- 1 root root  887 jui 21 10:20 hostkey.pem
```

La signature du certificat du hôte, se fait par l'utilisateur globus de la machine 4, donc, nous devons déplacer les deux fichiers résultant hostcert_request.pem et usercert_request.pem, de la machine ‘poste’ à la machine ‘poste4’ et les signer par l’utilisateur globus :

```
[globus@poste4 ~]$ grid-ca-sign -in /tmp/hostcert_request.pem -out
/tmp/hostsigend.pem
```

Le résultat sera :

```
To sign the request
```

```
please enter the password for the CA key:
The new signed certificate is at: /home/globus/.globus/simpleCA//newcerts/05.pem
```

Après la signature, nous remplaçons le fichier signé hostsigned.pem par le fichier hostcert.pem de la machine ‘poste’. Nous plaçons hostsigned.pem sous ‘/tmp/’ et nous exécutons la commande:

```
[globus@poste]$ mv /tmp/hostsigned.pem /etc/ grid-security/hostcert.pem
```

Il faut s’assurer que les fichiers hostcert.pem, hoscert_request.pem et hostkey ont les priorités suivantes en ordre: 644, 644 et 400. Une priorité excessive ou manquante peut générer des erreurs de permission.

De même, nous demandons le certificat pour les autres utilisateurs de la machine ‘poste’

On lance la commande suivante sous l’utilisateur globus de la seconde machine :

```
[globus@poste ~]$ grid-cert-request -ca -force -cn globus_poste
```

Résultat:

```
nondefaultca=true
The available CA configurations installed on this host are:
1)  637244ab - /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/CN=Globus Simple CA
 Enter the index number of the CA you want to sign your cert request: 1
Using CA: 637244ab - /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/CN=Globus Simple CA
   /home/globus/.globus/usercert_request.pem already exists
   /home/globus/.globus/usercert.pem already exists
   /home/globus/.globus/userkey.pem already exists
A certificate request and private key is being created.
You will be asked to enter a PEM pass phrase.
This pass phrase is akin to your account password,
and is used to protect your key file.
If you forget your pass phrase, you will need to
obtain a new certificate.
Generating a 1024 bit RSA private key
...........++++++
....................++++++
writing new private key to '/home/globus/.globus/userkey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
```

```
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Level 0 Organization [Grid]:Level 0 Organizational Unit [GlobusTest]:Level 1 Organizational Unit [simpleCA-poste4.lri.net]:Level 2 Organizational Unit [lri.net]:Name (e.g., John M. Smith) []:
 A private key and a certificate request has been generated with the subject:
/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=globus_poste
If the CN=globus_poste is not appropriate, rerun this
script with the -force -cn "Common Name" options.
Your private key is stored in /home/globus/.globus/userkey.pem
Your request is stored in /home/globus/.globus/usercert_request.pem
Please e-mail the request to the Globus Simple CA ahmed.dib@lri-annaba.net
You may use a command similar to the following:
  cat /home/globus/.globus/usercert_request.pem | mail ahmed.dib@lri-annaba.net
Only use the above if this machine can send AND receive e-mail. if not, please
mail using some other method.
Your certificate will be mailed to you within two working days.
If you receive no response, contact Globus Simple CA at ahmed.dib@lri-annaba.net
```

Les trois fichiers ‘ .pem’ générés sont les suivants :

```
[globus@poste ~]$ ls -l /home/globus/.globus/
```

```
total 20
-rw-r--r-- 1 globus globus    0 jui 21 10:22 usercert.pem
-rw-r--r-- 1 globus globus 1415 jui 21 10:22 usercert_request.pem
-r-------- 1 globus globus  963 jui 21 10:22 userkey.pem
```

Après la génération du fichier usercert_request.pem, nous devons le signer sur la machine 4, nous plaçons usercert_request.pem dans ‘/tmp/’ du poste4 et nous lançons la commande :

```
[globus@poste4 ~]$ grid-ca-sign -in /tmp/usercert_request.pem -out
/tmp/usersigend.pem
```

Le résultat sera:

```
To sign the request
please enter the password for the CA key:
The new signed certificate is at: /home/globus/.globus/simpleCA//newcerts/07.pem
```

Nous plaçons le fichier signé signed.pem de la machine 4, dans /tmp/ sur la machine ‘poste’ et nous lançons la commande suivante sous globus:

```
[globus@ poste]# mv /tmp/signed.pem /home/globus/.globus/usercert.pem
```

De même, il faut vérifier que les fichiers usercert.pem, usercert_request.pem et userkey.pem ont respectivement les priorités suivantes : 644, 644 et 400.

Finalement, nous lançons la création du proxy avec la commande :

```
[globus@ poste/]$ grid-proxy-init –debug -verify
```

Résultat:

```
User Cert File: /home/globus/.globus/usercert.pem
User Key File: /home/globus/.globus/userkey.pem
Trusted CA Cert Dir: /etc/grid-security/certificates
Output File: /tmp/x509up_u501
Your identity: /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=globus_poste
Enter GRID pass phrase for this identity:
Creating proxy ...++++++++++++
............................++++++++++++
 Done
Proxy Verify OK
Your proxy is valid until: Mon Jul 21 22:50:58 2008
```

De même, on demande le certificat pour le deuxième utilisateur 'saida' :

```
[saida@poste ~]$ grid-cert-request -ca
```

Le résultat:

```
nondefaultca=true
The available CA configurations installed on this host are:
1)  637244ab - /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/CN=Globus Simple CA
Enter the index number of the CA you want to sign your cert request: 1
Using CA: 637244ab - /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/CN=Globus Simple CA
A certificate request and private key is being created.
You will be asked to enter a PEM pass phrase.
This pass phrase is akin to your account password,
and is used to protect your key file.
If you forget your pass phrase, you will need to
obtain a new certificate.
Generating a 1024 bit RSA private key
................++++++
............................++++++
writing new private key to '/home/saida/.globus/userkey.pem'
Enter PEM pass phrase:
```

```
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Level 0 Organization [Grid]:Level 0 Organizational Unit [GlobusTest]:Level 1 Organizational Unit [simpleCA-poste4.lri.net]:Level 2 Organizational Unit [lri.net]:Name (e.g., John M. Smith) []:
A private key and a certificate request has been generated with the subject:
/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=root
If the CN=root is not appropriate, rerun this
script with the -force -cn "Common Name" options.
Your private key is stored in /home/saida/.globus/userkey.pem
Your request is stored in /home/saida/.globus/usercert_request.pem
Please e-mail the request to the Globus Simple CA ahmed.dib@lri-annaba.net
You may use a command similar to the following:
  cat /home/saida/.globus/usercert_request.pem | mail ahmed.dib@lri-annaba.net
Only use the above if this machine can send AND receive e-mail. if not, please
mail using some other method.
Your certificate will be mailed to you within two working days.
If you receive no response, contact Globus Simple CA at ahmed.dib@lri-annaba.net
Visualisation des trois fichiers '.pem' :
[saida@poste ~]$ ls -l /home/saida/.globus/
total 20
-rw-r--r-- 1 saida saida    0 jui 21 10:25 usercert.pem
-rw-r--r-- 1 saida saida 1408 jui 21 10:25 usercert_request.pem
-r-------- 1 saida saida  963 jui 21 10:25 userkey.pem
```

Après la génération du fichier hostcert_request.pem, nous devons le signer sur la machine 4, nous plaçons usercert_request.pem dans '/tmp/'

```
[globus@poste ~]$ cp /home/globus/.globus/usercert_request.pem /tmp
```

Et nous lançons la commande de signature:

```
[globus@poste4 ~]$ grid-ca-sign -in /tmp/usercert_request_user.pem -out
/tmp/usersigend_user.pem -force
```

Le résultat sera:

```
To sign the request
please enter the password for the CA key:
The new signed certificate is at: /home/globus/.globus/simpleCA//newcerts/06.pem
```

Maintenant nous allons copier le fichier signé usersigned_user.pem dans le fichier usercert.pem qui est vide en tapant la commande suivante :

```
[saida@poste ~]$ cp /tmp/usersigend_user.pem
/home/saida/.globus/usercert.pem
```

Visualisation des propriétés du fichier *usercert.pem :*

```
[saida@poste ~]$ ls -l /home/saida/.globus/usercert.pem
```

Résultat:

```
-rw-r--r-- 1 saida saida 2683 jui 21 10:50 /home/saida/.globus/usercert.pem
```

Finalement, nous lançons la création du proxy en tapant la commande suivante :

```
[saida@poste ~]$ grid-proxy-init -debug -verify
```

Résultat:

```
User Cert File: /home/saida/.globus/usercert.pem
User Key File: /home/saida/.globus/userkey.pem
Trusted CA Cert Dir: /etc/grid-security/certificates
Output File: /tmp/x509up_u500
Your identity: /O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=root
Enter GRID pass phrase for this identity:
Creating proxy .....++++++++++++
......++++++++++++
 Done
Proxy Verify OK
Your proxy is valid until: Mon Jul 21 22:55:09 2008
```

b) Ajout des autorisations

Comme précédemment, on crée le fichier grid-mapfile au niveau de la machine ‘poste’, et on ajoute deux entrées pour les deux utilisateur :’globus et saida’ :

```
[root@poste ~]# /usr/local/globus-4.0.6/sbin/grid-mapfile-add-entry -dn
"/O=Grid/OU=GlobusTest/OU=simpleCA-
poste4.lri.net/OU=lri.net/CN=globus_poste" -ln globus
```

```
Modifying /etc/grid-security/grid-mapfile ...
New entry:
"/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=globus_poste " globus
(1) entry added
```

Et pour l'utilisateur 'saida' comme suit

```
[root@poste ~]# /usr/local/globus-4.0.6/sbin/grid-mapfile-add-entry -dn
"/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=root" -ln
saida
```

```
Modifying /etc/grid-security/grid-mapfile ...
New entry:
"/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=root" saida
(1) entry added
```

c) Ajout des autorisations inter-postes :

Cette étape se traduit par l'ajout des entrées pour chaque utilisateur d'un poste i dans un autre j, on donne un exemple qui permet à l'utilisateur 'globus' de la machine 'poste' d'être reliée avec le poste 4, on ajoute au map du poste 4 l'entrée composée du sujet de certificat de l'utilisateur globus de l'hote 'poste' et de l'utilisateur globus de l'hote 'poste4'

```
[root@poste4 ~]# /usr/local/globus-4.0.6/sbin/grid-mapfile-add-entry -dn
"/O=Grid/OU=GlobusTest/OU=simpleCA-
poste4.lri.net/OU=lri.net/CN=globus_poste " -ln globus
```

De l'autre coté, on lance :

```
[root@poste ~]# /usr/local/globus-4.0.6/sbin/grid-mapfile-add-entry -dn
"/O=Grid/OU=GlobusTest/OU=simpleCA-poste4.lri.net/OU=lri.net/CN=globus " -
ln globus
```

5. SERVICE 'GRIDFTP':

5.1. Configuration du service 'gridFTP'

Nous configurons le service gridFTP pour qu'il se lance au démarrage, et pour se faire, nous utilisons xinetd ou inetd, dans notre cas, nous avons utilisé xinetd, si nous n'avons pas xinetd installé sur la machine, nous lançons son installation comme suit:

```
[root@poste4 ~]# yum install xinetd
```

Résultat:

```
Dependencies Resolved
=================================================================
 Package            Arch      Version        Repository      Size
=======================================================================
Installing:
 xinetd             i386      2:2.3.14-14.fc8  fedora          123 k
```

```
Transaction Summary
=============================================================================
Install      1 Package(s)
Update       0 Package(s)
Remove       0 Package(s)
Total download size: 123 k
Is this ok [y/N]: y
Downloading Packages:
(1/1): xinetd-2.3.14-14.f 100% |=========================| 123 kB    00:04
warning: rpmts_HdrFromFdno: Header V3 DSA signature: NOKEY, key ID 4f2a6fd2
Importing GPG key 0x4F2A6FD2 "Fedora Project <fedora@redhat.com>" from /etc/pki/rpm-gpg/RPM-GPG-KEY-fedora
Is this ok [y/N]: y
Importing GPG key 0xDB42A60E "Red Hat, Inc <security@redhat.com>" from /etc/pki/rpm-gpg/RPM-GPG-KEY
Is this ok [y/N]: y
Running rpm_check_debug
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing: xinetd                    ######################### [1/1]
Installed: xinetd.i386 2:2.3.14-14.fc8
Complete!
```

Nous créons un fichier 'gridftp' sous */etc/xinetd.d/*, avec la commande suivante:

```
[root@poste4 ~]#vim /etc/xinetd.d/gridftp
```

Et nous écrivons le contenu suivant:

```
service gsiftp
{
instances = 100
socket_type = stream
wait = no
user = root
env += GLOBUS_LOCATION = /usr/local/globus-4.0.6
env += LD_LIBRARY_PATH=/usr/local/globus-4.0.6/lib
server = /usr/local/globus-4.0.1/sbin/globus-gridftp-server
server_args = -i   root     70 jui 15 14:59 gridftp.conf
log_on_success += DURATION
nice = 10
disable = no
}
```

Nous lançons ou nous relançons xinetd comme suit:

```
[root@poste4 ~]# /etc/init.d/xinetd start
```

Résultat :

```
Démarrage de xinetd :                    [ OK ]
```

Ou avec

```
[root@poste4 ~] /etc/init.d/xinetd reload
```

Résultat :

```
Rechargement de la configuration :              [ OK ]
```

Nous ajoutons le service gsiftp à l'ensemble des services locaux, au niveau du fichier '/etc/services' :

Nous ajoutons la ligne suivante à la fin du fichier dans la section 'Local services'.

```
gsiftp        2811/tcp                  # GSI FTP
```

Une visualisation avec la commande 'tail' est possible:

```
[root@poste4 ~]# tail /etc/services

iqobject      48619/udp                 # iqobject
vboxd         20012/udp
binkp         24554/tcp                 # binkp fidonet protocol
asp           27374/tcp                 # Address Search Protocol
asp           27374/udp
dircproxy     57000/tcp                 # Detachable IRC Proxy
tfido         60177/tcp                 # fidonet EMSI over telnet
fido          60179/tcp
# Local services
gsiftp        2811/tcp                  # GSI FTP
```

Nous continuons la configuration de GridFTP en créant le fichier gridftp.conf sous '/etc/grid-security' et aussi sous '$GLOBUS_LOCATION/etc/'.

```
[root@poste4 ~]# vi /etc/grid-security/gridftp.conf
```

Et nous mettrons le contenu suivant:

```
port 2811
allow_anonymous 1
anonymous_user globus
banner "bienvenue!"
```

```
[root@poste4 ~]# cp /etc/grid-security/gridftp.conf /usr/local/globus-
4.0.6/etc/
```

Après la configuration de GridFTP, nous lançons le proxy pour l'utilisateur globus:

```
[globus@poste4 ~]$ grid-proxy-init -debug -verify
```

Le résultat sera:

```
User Cert File: /home/globus/.globus/usercert.pem
User Key File: /home/globus/.globus/userkey.pem
Trusted CA Cert Dir: /etc/grid-security/certificates
Output File: /tmp/x509up_u500
Your identity: /O=Grid/OU=GlobusTest/OU=simpleCA-poste2.lri.annaba/OU=lri.annaba/CN=globus
Enter GRID pass phrase for this identity:
Creating proxy .....++++++++++++
.++++++++++++
 Done
Proxy Verify OK
Your proxy is valid until: Tue Jul 15 00:58:16 2008
```

5.2. Lancement du service 'gridFTP'

De même, nous pouvons lancer manuellement le serveur gridFTP comme suit:

```
[root@poste4 ~]# cd /usr/local/globus-4.0.6/sbin/

[root@poste4 sbin]# ./globus-gridftp-server -c ./gridftp.conf
```

Résultat:

```
Server listening at poste2.lri.net:56312
```

Le port 56312 est choisi pour le serveur gridftp, nous pouvons personnaliser le port en ajoutant l'option « -p numéro de port »;

Exemple:

[root@poste2 sbin]# ./globus-gridftp-server -c ./gridftp.conf -p 5000.

Maintenant, nous pouvons utiliser le protocole gsiftp pour l'envoi des fichiers, par exemple :

```
[globus@poste4 ~]$ globus-url-copy -stripe
gsiftp://poste4.lri.net:56312/tmp/gridftp_test file:///tmp/copie_gridftp_test
```

5.3. Lancement du container des services web

Nous créons le script permettant le lancement du container, sous globus, nous exécutons la commande suivante:

```
[globus@poste4 ~]$ vim $GLOBUS_LOCATION/start-stop
```

Et nous plaçons le contenu suivant:

```
#!/bin/sh
set -e
export GLOBUS_OPTIONS="-Xms256M -Xmx512M"
sh $GLOBUS_LOCATION/etc/globus-user-env.sh
cd $GLOBUS_LOCATION
case "$1" in
start)
$GLOBUS_LOCATION/sbin/globus-start-container-detached -p 8443
;;
stop)
$GLOBUS_LOCATION/sbin/globus-start-container-detached
;;
*)
echo "Usage : globus start|stop">&2
exit 1
;;
esac
exit 0
```

Nous donnons le droit d'exécution pour ce fichier avec la commande :

```
[root@poste4 ~]$ chmod +x /usr/local/globus-4.0.6/start-stop
```

Sous root, nous créons un deuxième script sous '/etc/init.d/' qui va être utilisé et qui va faire appel au premier.

```
[root@poste4 ~]# vim /etc/init.d/globus-4.0.6
```

Et nous écrivons le contenu suivant:

```
#!/bin/sh -e
case "$1" in
start)
su - globus /usr/local/globus-4.0.6/start-stop start
;;
stop)
su - globus /usr/local/globus-4.0.6/start-stop stop
;;
restart)
$0 stop
sleep 1
$0 start
;;
*)
printf "usage : $0 start|stop|restart|n">&2
exit 1
;;
esac
exit 0
```

Nous donnons plus de priorité pour ce script ainsi le droit d'exécution avec la commande :

```
[root@poste4 ~]# chmod o+x /etc/init.d/globus-4.0.6
```

Maintenant, nous lançons le container comme suit:

```
[root@poste4 ~]# /etc/init.d/globus-4.0.6 start

Starting Globus container. PID: 3554
```

Pour vérifier la réussite du lancement, nous visualisons le fichier « container.log », en exécutant la commande suivante:

```
[root@poste4 ~]# vim /usr/local/globus-4.0.6/var/container.log
```

Le résultat sera :

```
2008-07-13 11:35:37,100 ERROR service.ReliableFileTransferImpl [main,<init>:69] Unable to setup database driver with pooling.Connection refused. Check that the hostname and port are correct and that the postmaster is accepting TCP/IP connections.

2008-07-13 11:35:37,550 WARN  service.ReliableFileTransferHome [main,initialize:97] All RFT requests will fail and all GRAM jobs that require file staging will fail.Connection refused. Check that the hostname and port are correct and that the postmaster is accepting TCP/IP connections.

Starting SOAP server at: https://192.168.0.103:8443/wsrf/services/

With the following services:

[1]: https://192.168.0.103:8443/wsrf/services/AdminService

[2]: https://192.168.0.103:8443/wsrf/services/AuthzCalloutTestService

[3]: https://192.168.0.103:8443/wsrf/services/CASService

[4]: https://192.168.0.103:8443/wsrf/services/ContainerRegistryEntryService

[5]: https://192.168.0.103:8443/wsrf/services/ContainerRegistryService

[6]: https://192.168.0.103:8443/wsrf/services/CounterService

[7]: https://192.168.0.103:8443/wsrf/services/DefaultIndexService

[8]: https://192.168.0.103:8443/wsrf/services/DefaultIndexServiceEntry

[9]: https://192.168.0.103:8443/wsrf/services/DefaultTriggerService

[10]: https://192.168.0.103:8443/wsrf/services/DefaultTriggerServiceEntry

[11]: https://192.168.0.103:8443/wsrf/services/DelegationFactoryService

[12]: https://192.168.0.103:8443/wsrf/services/DelegationService

[13]: https://192.168.0.103:8443/wsrf/services/DelegationTestService

[14]: https://192.168.0.103:8443/wsrf/services/InMemoryServiceGroup

[15]: https://192.168.0.103:8443/wsrf/services/InMemoryServiceGroupEntry

[16]: https://192.168.0.103:8443/wsrf/services/InMemoryServiceGroupFactory

[17]: https://192.168.0.103:8443/wsrf/services/IndexFactoryService

[18]: https://192.168.0.103:8443/wsrf/services/IndexService

[19]: https://192.168.0.103:8443/wsrf/services/IndexServiceEntry
```

```
[20]: https://192.168.0.103:8443/wsrf/services/ManagedExecutableJobService
[21]: https://192.168.0.103:8443/wsrf/services/ManagedJobFactoryService
[22]: https://192.168.0.103:8443/wsrf/services/ManagedMultiJobService
[23]: https://192.168.0.103:8443/wsrf/services/ManagementService
[24]: https://192.168.0.103:8443/wsrf/services/NotificationConsumerFactoryService
[25]: https://192.168.0.103:8443/wsrf/services/NotificationConsumerService
[26]: https://192.168.0.103:8443/wsrf/services/NotificationTestService
[27]: https://192.168.0.103:8443/wsrf/services/PersistenceTestSubscriptionManager
[28]: https://192.168.0.103:8443/wsrf/services/ReliableFileTransferFactoryService
[29]: https://192.168.0.103:8443/wsrf/services/ReliableFileTransferService
[30]: https://192.168.0.103:8443/wsrf/services/RendezvousFactoryService
[31]: https://192.168.0.103:8443/wsrf/services/ReplicationService
[32]: https://192.168.0.103:8443/wsrf/services/SampleAuthzService
[33]: https://192.168.0.103:8443/wsrf/services/SecureCounterService
[34]: https://192.168.0.103:8443/wsrf/services/SecurityTestService
[35]: https://192.168.0.103:8443/wsrf/services/ShutdownService
[36]: https://192.168.0.103:8443/wsrf/services/SubscriptionManagerService
[37]: https://192.168.0.103:8443/wsrf/services/TestAuthzService
[38]: https://192.168.0.103:8443/wsrf/services/TestRPCService
[39]: https://192.168.0.103:8443/wsrf/services/TestService
[40]: https://192.168.0.103:8443/wsrf/services/TestServiceRequest
[41]: https://192.168.0.103:8443/wsrf/services/TestServiceWrongWSDL
[42]: https://192.168.0.103:8443/wsrf/services/TriggerFactoryService
[43]: https://192.168.0.103:8443/wsrf/services/TriggerService
[44]: https://192.168.0.103:8443/wsrf/services/TriggerServiceEntry
[45]: https://192.168.0.103:8443/wsrf/services/Version
[46]: https://192.168.0.103:8443/wsrf/services/WidgetNotificationService
[47]: https://192.168.0.103:8443/wsrf/services/WidgetService
[48]: https://192.168.0.103:8443/wsrf/services/gsi/AuthenticationService
[49]: https://192.168.0.103:8443/wsrf/services/mds/test/execsource/IndexService
[50]: https://192.168.0.103:8443/wsrf/services/mds/test/execsource/IndexServiceEntry
[51]: https://192.168.0.103:8443/wsrf/services/mds/test/subsource/IndexService
[52]: https://192.168.0.103:8443/wsrf/services/mds/test/subsource/IndexServiceEntry
```

Les erreurs existantes, vont être corrigées après la configuration de la base de données.

6. CONFIGURATION DU RFT

6.1. Création du fichier pg_hba.conf

Le fichier pg_hba.conf est un fichier de configuration pour postgressql, il va contenir par la suite l'entrée qui autorise à l'utilisateur globus d'utiliser la base de données rftDatabase à partir du

poste4.

Nous créons le chemin suivant *'/var/lib/pgsql/data'*

```
[root@poste4 ~]# mkdir  /var/lib/pgsql/

[root@poste4 ~]# mkdir  /var/lib/pgsql/data/
```

On édite le fichier pg_hba.conf

```
[root@poste4 ~]# vim /var/lib/pgsql/data/pg_hba.conf
```

Nous ajoutons la ligne suivante :

```
host rftDatabase globus 192.168.0.103 255.255.255.0 md5
```

6.2. Création d'un utilisateur globus sous postgres

Nous lançons le serveur postgressql :

```
[postgres@poste4 ~]$ /usr/local/pgsql/bin/postmaster -i -D
/usr/local/pgsql/data/
```

Le résultat est le suivant:

```
LOG:  database system was interrupted at 2008-07-13 13:09:54 CET
LOG:  checkpoint record is at 0/42E904
LOG:  redo record is at 0/42E904; undo record is at 0/0; shutdown FALSE
LOG:  next transaction ID: 0/622; next OID: 24579
LOG:  next MultiXactId: 1; next MultiXactOffset: 0
LOG:  database system was not properly shut down; automatic recovery in progress
LOG:  record with zero length at 0/42E94C
LOG:  redo is not required
LOG:  database system is ready
```

Nous utilisons la base de données *test* créée précédemment :

```
[postgres@poste4 ~]$ /usr/local/pgsql/bin/psql test
Welcome to psql 8.2.6, the PostgreSQL interactive terminal.
Type:  \copyright for distribution terms
       \h for help with SQL commands
       \? for help with psql commands
       \g or terminate with semicolon to execute query
       \q to quit
test=#
```

Nous lançons la création de l'utilisateur globus sous postgres en lançant la requête suivante :

```
test=# CREATE USER globus WITH PASSWORD 'globus' CREATEDB;
```

Résultat :

```
CREATE ROLE
```

6.3. Création de la base de données rftDatabase :

Après la création de l'utilisateur, nous allons lancer la création de la base de données rftDatabase

```
[globus@poste4 ~]$ /usr/local/pgsql/bin/createdb rftDatabase
```

Résultat :

```
CREATE DATABASE
```

Nous exécutons le script rft_schema.sql qui nous permet de créer les schémas de la base de données

```
[globus@poste4 ~]$  /usr/local/pgsql/bin/psql -d rftDatabase -f
/usr/local/globus-4.0.6/share/globus_wsrf_rft/rft_schema.sql
```

Le résultat sera:

```
psql:/usr/local/globus-4.0.6/share/globus_wsrf_rft/rft_schema.sql:6: NOTICE:  CREATE TABLE / PRIMARY
KEY will create implicit index "requestid_pkey" for table "requestid"
CREATE TABLE
psql:/usr/local/globus-4.0.6/share/globus_wsrf_rft/rft_schema.sql:11: NOTICE:  CREATE TABLE / PRIMARY
KEY will create implicit index "transferid_pkey" for table "transferid"
CREATE TABLE
psql:/usr/local/globus-4.0.6/share/globus_wsrf_rft/rft_schema.sql:30: NOTICE:  CREATE TABLE / PRIMARY
KEY will create implicit index "request_pkey" for table "request"
CREATE TABLE
psql:/usr/local/globus-4.0.6/share/globus_wsrf_rft/rft_schema.sql:65: NOTICE:  CREATE TABLE / PRIMARY
KEY will create implicit index "transfer_pkey" for table "transfer"
CREATE TABLE
psql:/usr/local/globus-4.0.6/share/globus_wsrf_rft/rft_schema.sql:71: NOTICE:  CREATE TABLE / PRIMARY
KEY will create implicit index "restart_pkey" for table "restart"
CREATE TABLE
CREATE TABLE
CREATE INDEX
```

Nous vérifions les headers du fichier jndi-config.xml, et pour cela nous lançons la commande :

```
[globus@poste4 globus_wsrf_rft]$  grep -C 3 password
$GLOBUS_LOCATION/etc/globus_wsrf_rft/jndi-config.xml
```

Le résultat sera :

```
</parameter>
        <parameter>
          <name>
          password
          </name>
          <value>
          foo
```

6.4. Test de fonctionnement du RFT :

Avant de tester le bon fonctionnement du rft, nous devons relancer le container des web services avec l'option restart s'il est déjà lancé, sinon le lancer de nouveau avec l'option start.

```
[root@poste4 ~]# /etc/init.d/globus-4.0.6 restart
```

Le résultat sera :

```
Stale pid file detected. It will be removed
Starting Globus container. PID: 3554
```

Nous testons le bon fonctionnement du rft, nous lançons la commande qui fait référence au fichier runtests.xml

```
[globus@poste4 ~]$ ant -Dtests.jar=/usr/local/globus-
4.0.6/lib/globus_wsrf_rft_test.jar -f /usr/local/globus-
4.0.6/share/globus_wsrf_rft_test/runtests.xml
```

Le résultat sera comme suit:

```
Buildfile: /usr/local/globus-4.0.6/share/globus_wsrf_rft_test/runtests.xml
init:
  [delete] Deleting directory /usr/local/globus-4.0.6/share/globus_wsrf_rft_test/tests/classes
   [mkdir] Created dir: /usr/local/globus-4.0.6/share/globus_wsrf_rft_test/tests/classes
   [unjar] Expanding: /usr/local/globus-4.0.6/lib/globus_wsrf_rft_test.jar into /usr/local/globus-
4.0.6/share/globus_wsrf_rft_test/tests/classes
runServer:
runTests:
_runCustomTests:
   [junit] Running org.globus.transfer.reliable.service.test.PackageTests
   [junit]     GSSException: Expired credentials detected)
   [junit] Tests run: 18, Failures: 0, Errors: 17, Time elapsed: 11,096 sec
```

```
    [junit] Test org.globus.transfer.reliable.service.test.PackageTests FAILED
    [junit] Running org.globus.transfer.reliable.service.test.client.PackageTests
    [junit] Tests run: 2, Failures: 2, Errors: 0, Time elapsed: 0,322 sec
    [junit] Test org.globus.transfer.reliable.service.test.client.PackageTests FAILED
    [junit] Running org.globus.transfer.reliable.service.test.connection.PackageTests
    [junit] Tests run: 81, Failures: 0, Errors: 81, Time elapsed: 1,065 sec
    [junit] Test org.globus.transfer.reliable.service.test.connection.PackageTests FAILED
BUILD SUCCESSFUL
Total time: 15 seconds
```

Nous pouvons exécuter un nouveau test qui fait référence au fichier mentionné précédemment, en lançant la commande suivante :

```
[globus@poste2 ~]$ ant -f /usr/local/globus-
4.0.6/share/globus_wsrf_rft_test/runtests.xml generateTestReport
```

Le résultat sera:

```
Buildfile: /usr/local/globus-4.0.6/share/globus_wsrf_rft_test/runtests.xml
generateTestReport:
 [delete] Deleting directory /usr/local/globus-4.0.6/share/globus_wsrf_rft_test/tests/test-reports-html
 [mkdir] Created dir: /usr/local/globus-4.0.6/share/globus_wsrf_rft_test/tests/test-reports-html
[junitreport] Processing /usr/local/globus-4.0.6/share/globus_wsrf_rft_test/tests/test-reports/TESTS-
TestSuites.xml to /tmp/null6835105
[junitreport] Loading stylesheet jar:file:/usr/local/outils/apache-ant-1.7.0/lib/ant-
junit.jar!/org/apache/tools/ant/taskdefs/optional/junit/xsl/junit-frames.xsl
[junitreport] Transform time: 1050ms
[junitreport] Deleting: /tmp/null6835105
BUILD SUCCESSFUL p
Total time: 1 second
```

7. CONFIGURATION DU SERVICE GRAM

7.1. Édition du fichier '/etc/sudoers'

Après avoir installer gridFTP et le RFT, il est maintenant possible de lancer l'installation du gestionnaire des ressources GRAM, pour se faire, nous faisons l'édition du fichier '/etc/sudoers' :

```
[root@poste4 ~]# vim /etc/sudoers
```

Nous ajoutons :

```
globus ALL=(user1,user2) NOPASSWD: /usr/local/globus-4.0.6/libexec/globus-gridmap-and-execute
-g /etc/grid-security/grid-mapfile /usr/local/globus-4.0.6/libexec/globus-job-manager-script.pl *
globus  ALL=(user1,user2) NOPASSWD: /usr/local/globus-4.0.6/libexec/globus-gridmap-and-execute
-g /etc/grid-security/grid-mapfile /usr/local/globus-4.0.6/libexec/globus-gram-local-proxy-tool *
```

7.2. Test du fonctionnement de GRAM

Avec la modification apportée précédemment, nous pouvons lancer GRAM et soumettre des jobs, nous lançons la commande suivante :

```
[globus@poste4 ~]$ globusrun-ws -submit -c /bin/true
```

Le résultat sera donc:

```
Submitting job...Done.
Job ID: uuid:b39456d8-526b-11dd-a4f2-001ec92c2a43
Termination time: 07/16/2008 12:44 GMT
Current job state: Active
Current job state: CleanUp
Current job state: Done
Destroying job...Done.
```

8. CONFIGURATION DE GRIDFTP, RFT ET GRAM SOUS LES AUTRES MACHINES

Après l'installation du certificat sur les autres machines (machine, machine2 et machine3) voir (section 4.4) nous procédons à la configuration des autres services (gridFTP, lancement des web services, RFT, GRAM) de la même façon que pour la machine4 (serveur du CA).

9. CONCLUSION

L'installation de Globus est une tâche qui nécessite beaucoup de précision. Dans notre expérience, l'installation a pris plus de deux semaines et cela est du aux problèmes qui sont liés à l'authentification des utilisateurs utilisant le Grid. Nous constatons de cette expérience que si l'autorité de certification AC est bien installée, et que les fichiers '*cert.pem' et '*key.pem' sont bien définis, nous éviterons une grande partie des erreurs qui peuvent se produire pendant le processus d'installation. De l'autre coté, il faut toujours mettre à jour le fichier 'grid-mapfile' dans chaque nœud de Grid en lui ajoutant le sujet et le nom de chaque nouvel utilisateur qui accède au Grid et même les utilisateurs des autres machines qui appartiennent à la même autorité. En fin, Pour mieux tester, vérifier, et tirer des conclusions sur les performances de l'intergiciel, une application doit être programmée puis exécutée sur le Grid.